Theses and Dissertations   1. Thesis and Dissertation Collection, all items

2000

Graphical user interface for a physical optics radar cross section prediction code

Garrido, Elmo E.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/32958

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

GRAPHICAL USER INTERFACE FOR A PHYSICAL
OPTICS RADAR CROSS SECTION PREDICTION CODE

by

Elmo E. Garrido, Jr.

September 2000

Thesis Advisor: David C. Jenn
Second Reader: D. Curtis Schleher

Approved for public release; distribution is unlimited

20001026 077

REPORT DOCUMENTATION PAGE		*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE September 2000	3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE : Graphical User Interface For A Physical Optics Radar Cross Section Prediction Code		**5. FUNDING NUMBERS**
6. AUTHOR(S) Garrido, Elmo Jr. E.		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)		**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	**12b. DISTRIBUTION CODE**

13. ABSTRACT *(maximum 200 words)*
    This implementation of the physical optics approximation method for predicting the radar cross section (RCS) of complex objects utilizes the scientific computational features of MATLAB and its Graphical User Interface (GUI) functions to provide an error-free encoding of input parameters and efficient calculation. The POFACETS GUI provides a convenient tool for both students and electromagnetic professionals to design complex models by representing its component parts as triangular facets and offers options for defining the surface roughness of the scattering object. The GUI calculates the monostatic or bistatic radar cross section of the modeled object based on certain radar frequencies and parameters of interest, creates a variety of plots for the model geometry and its RCS measurement, and provides additional functionalities to enhance visualization of the object and check for accuracy of data. Various help windows and messages are available to the user to serve as guide in the efficient use of the GUI. The simplicity of the GUI design does not require the user to have a detailed understanding of the various steps involved in the RCS calculation.

13. SUBJECT TERMS Physical Optics, Radar Cross Section, Monostatic, Bistatic, Electromagnetic Scattering, Graphical User Interface	**15. NUMBER OF PAGES** 89
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

GRAPHICAL USER INTERFACE FOR A PHYSICAL OPTICS RADAR CROSS SECTION PREDICTION CODE

Elmo E. Garrido, Jr.
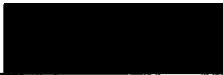Commander, Philippine Navy
B.S., Philippine Military Academy, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

from the
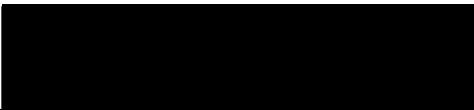
NAVAL POSTGRADUATE SCHOOL
September 2000

Author: _____
Elmo E. Garrido, Jr.

Approved by: _____
David C. Jenn, Thesis Advisor

_____
D. Curtis Schleher, Second Reader

_____
Dan C. Boger, Chairman
Information Warfare Academic Group

iii

ABSTRACT

This implementation of the physical optics approximation method for predicting the radar cross section (RCS) of complex objects utilizes the scientific computational features of MATLAB and its Graphical User Interface (GUI) functions to provide an error-free encoding of input parameters and efficient calculation. The POFACETS GUI provides a convenient tool for both students and electromagnetic professionals to design complex models by representing its component parts as triangular facets and offers options for defining the surface roughness of the scattering object. The GUI calculates the monostatic or bistatic radar cross section of the modeled object based on certain radar frequencies and parameters of interest, creates a variety of plots for the model geometry and its RCS measurement, and provides additional functionalities to enhance visualization of the object and verify correctness of data. Various help windows and messages are available to the user to serve as guide in the efficient use of the GUI. The simplicity of the GUI design does not require the user to have a detailed understanding of the various steps involved in the RCS calculation.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I. INTRODUCTION

Designing operational platforms and predicting their radar cross section (RCS) for a range of frequencies provides a challenging problem in modern electronic warfare. One approach to high frequency prediction calculations is to estimate a complex model with an array of simple shapes, such as triangles or flat plates. The RCS is obtained by computing the scattered field of the collection of these simple shapes to obtain the total RCS of the target.

Among the most common RCS prediction methods for any arbitrary three-dimensional target are the Method of Moments (MOM), the Finite Difference Method (FDM), Geometrical Optics (GO) and Physical Optics (PO). The MOM is a popular technique that solves the electric or magnetic field integral equation by reducing it to a matrix problem with the matrix size directly related to the electrical size of the scattering body in terms of the incident wave. This method gives very accurate results but is computationally intense. In the FDM, finite differences are used to approximate the differential operators in Maxwell's equations in either the time or frequency domain. The target must be discretized as in the MOM. Since this method necessarily calculates the fields in the region around the target, computing the RCS of a target with a characteristic dimension of several orders of magnitude of the wavelength would entail considerable time. GO and the Geometric Theory of Diffraction (GTD) involve ray tracing methods to analyze electrically large targets of arbitrary shape. Diffracted rays are allowed to originate from the scattering of the incident wave at edges, corners, and vertices. Among the problems associated with GO is that the source and the observation point must satisfy

Snell's Law or the reflected field is zero. It yields discontinuous fields at shadow boundaries and zero fields in the shadow regions. GTD solves this problem by accounting for the edge-related effects. PO overcomes the problems associated with the infinities of ray tracing for flat and singly curved surfaces by approximating the induced surface fields. The induced currents are integrated over the illuminated portions of the target to obtain the scattered far field while setting the current to zero over the shadowed portions. It is most accurate in the specular direction and gives good estimates at points far from discontinuities and shadow boundaries. However, because PO abruptly sets the current to zero at a shadow boundary, the computed field values at wide angles and in the shadow regions are inaccurate. Also, surface waves, multiple reflections and edge diffractions are not included [Refs. 1, 2 and 3].

There is a need for an easy-to-use tool that conveniently allows the modeling of an arbitrary three-dimensional object with component triangular facets, visualization of the geometry, and calculation of the radar cross section. The physical optics approximation is an efficient method that gives accurate results for most large targets, yet is not too computationally demanding. To make this tool appealing to the user and prevent, if not eliminate unnecessary errors in the setting up of various parameters, the design of a graphical user interface is desirable.

This thesis focuses on the design, development, and implementation of a Graphical User Interface (GUI) in MATLAB for a PO-based RCS prediction code. The program allows creation of a model comprised of triangular facets with options for specifying the surface characteristics. It then calculates the monostatic or bistatic radar

2

cross sections given certain computational parameters without requiring the user to have a detailed understanding of the various steps involved in the calculation.

The end product of this thesis is the POFACETS GUI that implements a physical optics approximation method that may be used by both students and electromagnetic professionals as a convenient tool to design complex models and predict their RCS performance based on the radar frequency and other parameters of interest. The GUI provides additional functionalities to enhance visualization of the model geometry and generates a variety of plots to present the RCS data. The GUI is based on codes developed by Professor David Jenn and utilizes the scientific computational features of MATLAB.

Chapter II gives a brief description of the theory behind the formulation of the code by providing an overview of radar fundamentals, electromagnetic scattering and a comparative discussion of the common methods of high frequency RCS prediction.

In Chapter III the flow or framework of the code is presented with the supporting mathematics involved in using the physical optics method to calculate the RCS.

Chapter IV describes the POFACETS Graphical User Interface. It is intended to provide a descriptive guide in making the user acquainted with the operation of the GUI. A step-by-step illustration is provided for working on the examples that are discussed.

Finally, Chapter V summarizes the research and suggests further work that could be performed to enhance the code.

THIS PAGE INTENTIONALLY LEFT BLANK

II.   THEORETICAL FORMULATION

A.   RADAR BACKGROUND

The acronym radar was coined during World War II from the phrase radio detection and ranging. A radar is a device that transmits an electromagnetic wave and detects objects by virtue of the energy scattered from them in the direction of the receiver. Generally, the radar transmitter and receiver can be at different locations when viewed from the target and this is called a bistatic radar. In most applications the transmitter and receiver are co-located and often share the same antenna, a configuration referred to as a monostatic radar. The classical form of the radar equation is

$$P_r = (\frac{P_t G_t}{4\pi R^2})(\frac{\sigma}{4\pi R^2})(\frac{G_r \lambda^2}{4\pi})L \qquad (2.1)$$

where P_r is the power received (in watts), P_t the output power of the transmitter, G_t the gain of the transmitter, G_r the gain of the receiver, σ the radar cross section (in square meters), λ the wavelength (in meters), L the reduction in the received signal due to losses $(0 \le L \le 1)$, and R the range of the target from the radar. The first term in parenthesis is the power density at the target (in watts/m^2) due to radiation from the transmitter. The second term is the incident power reflected or scattered back to the receiver. The third term is the amount of the reflected power which is captured by the receiving antenna due to its effective area. For the case of a monostatic radar with the same antenna used for transmitting and receiving (G_t is the same as G_r), equation 2.1 simplifies to

$$P_r = \frac{P_t G^2 \sigma \lambda^2 L}{(4\pi)^3 R^4} \qquad (2.2)$$

5

B.   ELECTROMAGNETIC SCATTERING

1.   Radar Cross Section

Radar cross section (RCS) is a measure of the power scattered in a given direction when a target is illuminated by an incident wave, normalized to the power density of the incident field. The purpose of the normalization is to remove the effect of the range and arrive at an RCS computation that is independent of the distance between the target and illumination source.   In terms of the incident and scattered electric field intensities, $\vec{E}_i$ and $\vec{E}_s$, the RCS is defined as

$$\sigma = \lim_{R \to \infty} 4\pi R^2 \frac{\left|\vec{E}_s\right|^2}{\left|\vec{E}_i\right|^2} \tag{2.3}$$

The unit of RCS most commonly used is decibels relative to a square meter (dBsm):

$$\sigma, \text{dBsm} = 10\log(\sigma, \text{m}^2) \tag{2.4}$$

Radar cross section is a function of many factors which include the target configuration and its material composition, frequency or wavelength, transmitter and receiver polarizations, and the target aspect (angular orientation of the target) relative to the radar. Hence, in general, σ could be specified as σ_{pq} (θ, ϕ) where q and p pertain to the incident and received polarizations, respectively, and where (θ, ϕ) are the polar spherical angles of view as shown in Figure 1.

6
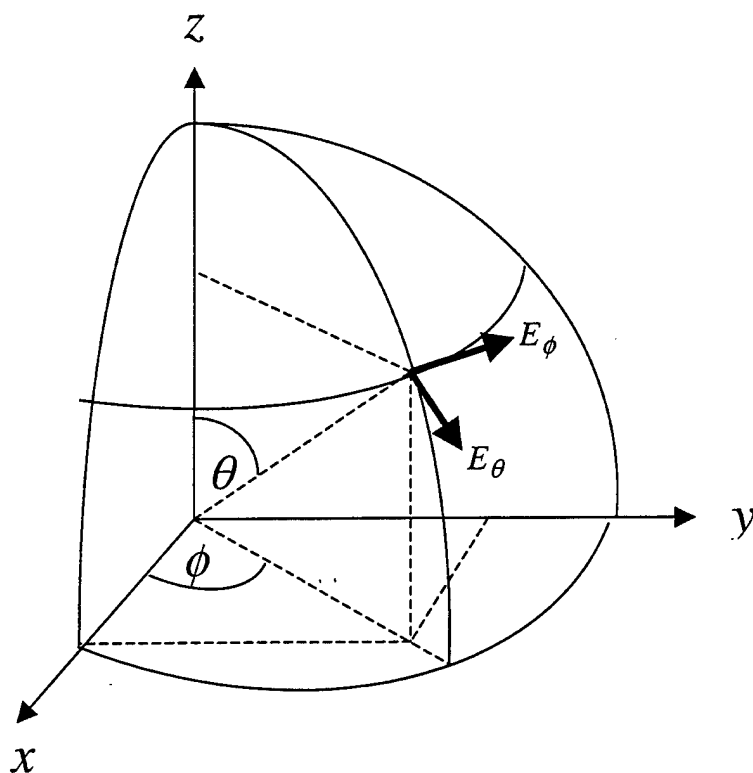
Figure 1. Coordinate System for Incident Polarization

2.   Scattering Regions

The scattering characteristics of a target are dependent on the frequency of the incident wave. Three regimes distinctly differentiate the RCS scattering of a target depending on the ratio of the wavelength λ to the body size L. The first is the low frequency region or *Rayleigh region* ($\lambda \gg L$) where the phase variation of the incident plane wave across the target is small and the shape of the body is not of importance. The second is the resonance region or *Mie region* ($\lambda \approx L$) where the phase variation of the current across the body is significant and all parts contribute to the scattering pattern. The third is the high frequency region or *Optical region* ($\lambda \ll L$) where the phase of the current across the body varies with many cycles and the scattering process can often be

7

considered as the summation of the returns from non-interacting isolated scattering centers [Refs. 1, 2 and 4].

3. Target Scattering Matrix

The radar cross section is a scalar number that is a function of the polarization of the incident and received waves. The polarization of the incident wave from the radar will not always be in the same orientation as the polarization of the target coordinate system and thus, it is necessary to decompose the polarization of the incident wave into components in the target coordinate system. Since the wave vector components at large distances are tangent to a sphere, two orthogonal components in terms of the variables θ and ϕ are sufficient to represent the incident field in a spherical system centered at the target. Thus, in general, the incident field can be written as

$$\bar{E}_i = E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi} \qquad (2.5)$$

where $\hat{\theta}$ and $\hat{\phi}$ are unit vectors in the target coordinate system. The polarization of the scattered field is not necessarily the same as the incident field since most complex targets generate a cross-polarized scattering component due to multiple diffractions and reflections. The following scattering matrix can be used to specify the relationship between the incident and scattered fields:

$$\begin{bmatrix} E_{s\theta} \\ E_{s\phi} \end{bmatrix} = \begin{bmatrix} S_{\theta\theta} & S_{\theta\phi} \\ S_{\phi\theta} & S_{\phi\phi} \end{bmatrix} \begin{bmatrix} E_{i\theta} \\ E_{i\phi} \end{bmatrix} \qquad (2.6)$$

The S_{pq} are the scattering parameters, where the first index specifies the polarization of the receive antenna and the second, the polarization of the incident wave. The elements of the scattering matrix are complex quantities and in terms of the RCS [Ref. 1],

$$S_{pq} = \frac{\sqrt{\sigma_{pq}}}{\sqrt{4\pi R^2}} e^{j\Psi_{pq}}, \quad \Psi_{pq} = \tan^{-1}\left\{ \frac{\text{Im}(E_{sp}/E_{iq})}{\text{Re}(E_{sp}/E_{iq})} \right\} \tag{2.7}$$

B.   COMMON RCS PREDICTION METHODS

Among the most common numerical RCS prediction methods for any arbitrary three-dimensional target are the Method of Moments (MOM), the Finite Difference Method (FDM), Geometrical Optics (GO) and Physical Optics (PO). The first two provide rigorous solutions but are computationally intense requiring considerable computer time to calculate results, while the latter two are approximate methods which give accurate and fast results, but have limitations.

1.   Method of Moments

The MOM involves breaking the target into a number of surface or line subdomains. This popular solution technique solves the electric or magnetic field integral equation by reducing it to a matrix problem with the matrix size directly related to the electrical size of the scattering body in terms of the incident wavelength. The coefficients of a series approximation to the current can then be determined using matrix algebra. In general, the subdomains must be small compared to the wavelength. This method gives very accurate solutions and is used in programs such as the Numerical Electromagnetics Code (NEC) [Ref. 3]. However, the calculation time is dictated by that required to solve the simultaneous equations and the method may not be practical for large targets at high frequencies [Refs. 1 and 3].

2. Finite Difference Methods

In this method, finite differences are used to approximate the differential operators in Maxwell's equations, usually in the time domain. The target must be discretized as in the MOM and the elements have to be small compared to the wavelength. Finite difference techniques do not require the solution of large dense matrices that the MOM does because the solution is stepped in time throughout the scattering body. Since this method calculates the fields in a computational grid around the target, calculating the RCS of a target with a characteristic dimension of several orders of magnitude of the wavelength would entail considerable time to compute [Refs. 1 and 3].

3. Geometrical Optics

GO involves ray tracing methods to analyze electrically large targets of arbitrary shape. The rules for ray tracing in a simple medium (linear, homogeneous, and isotropic) are similar to reflection and refraction in optics. In addition, diffracted rays are allowed that originate from the scattering of the incident wave at edges, corners, and vertices. The formulas are derived on the basis of infinite frequency $(\lambda \to 0)$ which implies an electrically large target. Among the problems associated with GO is that the source and the observation points must satisfy Snell's Law or the reflected field is zero. It yields discontinuous fields at shadow boundaries and zero fields in the shadow regions. The Geometric Theory of Diffraction was developed to take into account the edge-related effects by decomposing the total field at the observation point into its GO and diffracted components, thereby resulting in approximations for edge diffracted fields [Refs. 1 and 4].

10

4. Physical Optics

Physical Optics overcomes the problems associated with the infinities of ray tracing for flat and singly curved surfaces by approximating the induced surface current on the illuminated portions of the body to be proportional to the incident magnetic field intensity and integrating them to obtain the scattered far field. On the shadowed portion of the target, the current is set to zero. Thus, the current is approximated as

$$\vec{J}_s \approx \begin{cases} 2\hat{n} \times \vec{H}_i \, , & \text{for the illuminated portion} \\ 0 \, , & \text{for the shadowed portion} \end{cases} \tag{2.8}$$

where $\vec{H}_i$ is the incident magnetic field intensity at the surface and $\hat{n}$ is the surface normal unit vector.

The approximation in equation (2.8) is used in the radiation integrals to compute the scattered far field from the target. PO is a high-frequency approximation method that gives best results for electrically large bodies $(L \geq 10\lambda)$. It is most accurate in the specular direction and gives good estimates at points far from discontinuities and shadow boundaries. However, because PO abruptly sets the current to zero at a shadow boundary, the computed field values at wide angles and in the shadow regions are inaccurate. Also, surface waves, multiple reflections and edge diffractions are not included [Refs. 1, 2 and 4].

THIS PAGE INTENTIONALLY LEFT BLANK

III. PHYSICAL OPTICS APPLIED TO TRIANGULAR PATCHES

A. SCATTERING FROM FACETS WITH ARBITRARY ORIENTATION

The geometry of a single facet with an arbitrary orientation is shown in Figure 2. The facet vertices are known in terms of their cartesian coordinates (x_n, y_n, z_n) for $n = 1...3$. It is assumed that the vertices are labeled in a right-hand sense (counterclockwise) so that the thumb of the right hand gives the outward normal direction. For a closed body that surrounds the origin, the outward normal will in general point away from the origin. The position vectors to the nodes are given by $\vec{r}_n = x_n \hat{x} + y_n \hat{y} + z_n \hat{z}$.
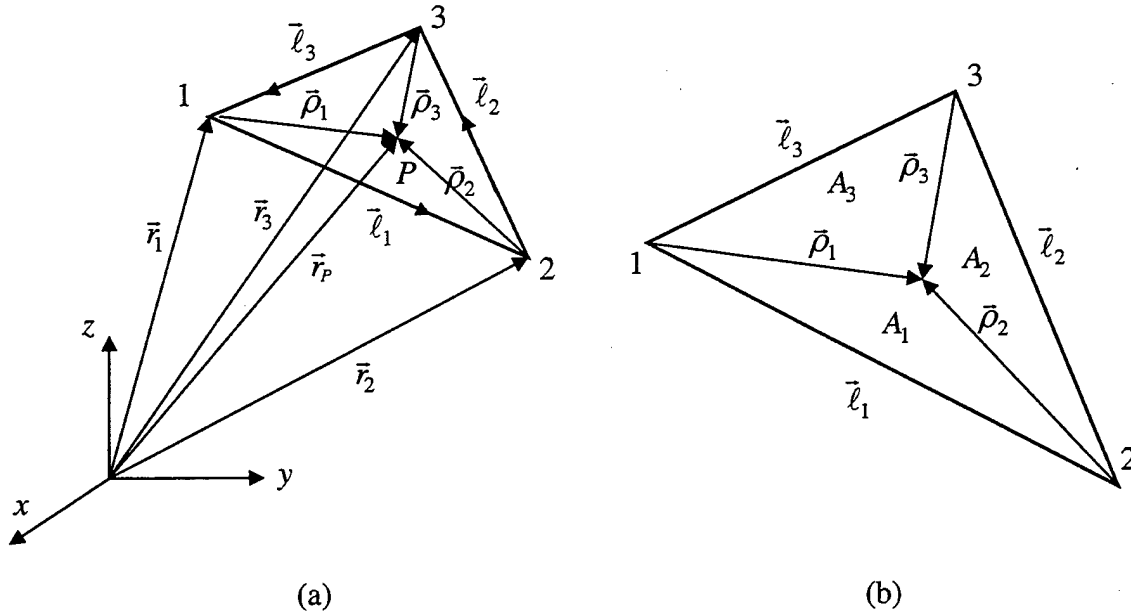
Figure 2. Arbitrarily Oriented Facet. (a) Arbitrarily Oriented Triangular Facet, (b) Facet Sub-Areas for Natural Coordinates

The edge vectors are determined from the node coordinates by

$$\vec{\ell}_1 = \vec{r}_2 - \vec{r}_1, \quad \vec{\ell}_2 = \vec{r}_3 - \vec{r}_2, \quad \vec{\ell}_3 = \vec{r}_1 - \vec{r}_3 \tag{3.1}$$

13

The tip of edge vector n originates at node n. The outward normal is obtained by taking the cross product of any two edge vectors in a right-handed sense. For example,

$$\hat{n} = \frac{\bar{\ell}_1 \times \bar{\ell}_3}{|\bar{\ell}_1||\bar{\ell}_3|} \equiv n_x \hat{x} + n_y \hat{y} + n_z \hat{z} \tag{3.2}$$

With this ordering of the vertices, the outward side of the facet is denoted as the *front face* and the opposite side as the *back face*. If a plane wave is incident from an angle (θ_i, ϕ_i) propagating towards the origin, then its propagation vector is given by

$$\hat{k}_i = -\hat{r} = -(\hat{x} \sin\theta_i \cos\phi_i + \hat{y} \sin\theta_i \sin\phi_i + \hat{z} \cos\theta_i) = -(\hat{x} u_i + \hat{y} v_i + \hat{z} w_i) \tag{3.3}$$

where (u_i, v_i, w_i) are the x, y and z direction cosines, respectively, and $\hat{r}$ the radial unit vector from the origin to the source at (θ_i, ϕ_i). It is a simple matter to determine whether the front face is illuminated or not by performing the test

$$-\hat{k}_i \bullet \bar{n} \geq 0 \tag{3.4}$$

The total area of the facet, A, can be computed from a vector relationship between edges

$$A = \frac{1}{2} |\bar{\ell}_1 \times \bar{\ell}_3| \tag{3.5}$$

In order to simplify integration over the surface of a triangle, normalized area coordinates (also known as simplex or natural coordinates) are used. Consider an integration point P located on the plate at (x_P, y_P, z_P). Its position vector is

$$\bar{r}_P = x_P \hat{x} + y_P \hat{y} + z_P \hat{z} \tag{3.6}$$

and

$$\bar{\rho}_1 = \bar{r}_P - \bar{r}_1, \quad \bar{\rho}_2 = \bar{r}_P - \bar{r}_2, \quad \bar{\rho}_3 = \bar{r}_P - \bar{r}_3 \tag{3.7}$$

14

Sub-areas of the total facet area can be defined, from which normalized area coordinates are computed

$$\zeta = \frac{A_1}{A}, \quad \xi = \frac{A_2}{A}, \quad \eta = \frac{A_3}{A} \tag{3.8}$$

Since $\zeta + \xi + \eta = 1$ because $A_1 + A_2 + A_3 = A$, then $\zeta = 1 - \xi - \eta$, and

$$\iint\limits_A ds_P = 2A \int\limits_0^1 \int\limits_0^{1-\eta} d\xi d\eta \tag{3.9}$$

B.   INCIDENT AND SCATTERED FIELDS

In general, the incident field is of the form

$$\vec{E}_i = (E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi})e^{-j\vec{k}_i \bullet \vec{r}}$$

$$\vec{H}_i = \frac{\vec{k}_i \times \vec{E}_i}{Z_o} = \frac{1}{Z_o}(E_{i\phi}\hat{\theta} - E_{i\theta}\hat{\phi})e^{-j\vec{k}_i \bullet \vec{r}} \tag{3.10}$$

where Z_O is the impedance of free space and $\vec{k}_i = k\hat{k}_i$, $(k = \frac{2\pi}{\lambda})$. The physical optics approximation for the surface current flowing on the facet is

$$\vec{J}_s = 2\hat{n} \times \vec{H}_i = \frac{2}{Z_O}\hat{n} \times (E_{i\phi}\hat{\theta} - E_{i\theta}\hat{\phi})e^{jkh} \tag{3.11}$$

where, for convenience, a variable h has been defined

$$h = \vec{r}_P \bullet \hat{r}_i = x_P u_i + y_P v_i + z_P w_i \tag{3.12}$$

The scattered field is found using the radiation integral

$$\vec{E}_s = \frac{jkZ_O}{4\pi r}e^{-jkr}\iint\limits_A \vec{J}_s e^{jkg} ds_P \tag{3.13}$$

where

$$g = \vec{r}_P \bullet \hat{r} = x_P u + y_P v + z_P w \tag{3.14}$$

15

Note that for the monostatic case

$$g = h \qquad (3.15)$$

The scattered field is written as

$$\vec{E}_s = \frac{jk}{2\pi r} e^{-jkr} \underbrace{\left[\hat{n} \times \left(E_{i\phi}\hat{\theta} - E_{i\theta}\hat{\phi}\right)\right]}_{\substack{\text{POLARIZATION} \\ \text{FACTOR, } PF}} \underbrace{\int\int_A e^{jk(h+g)} ds_P}_{\equiv I_c} \qquad (3.16)$$

It is not possible to obtain an exact closed form result for the integral. However, given that the incident wavefront is plane and known at the facet vertices, the amplitude and phase at interior integration points can be found by interpolation. Then the integrand can be expanded using a Taylor series, and each term integrated to give a closed form result. Usually a small number of terms (on the order of 5) will give a sufficiently accurate approximation for the integral.

Using the notation of [Ref. 5], define

$$I_c = \int\int_A C_c(\eta,\xi) e^{jD_c(\eta,\xi)} d\eta d\xi \qquad (3.17)$$

where

$$C_c(\eta,\xi) = C_p\eta + C_q\xi + C_o \equiv \left|\vec{E}_i\right| \qquad (3.18)$$

For a unit amplitude plane wave

$$\left|\vec{E}_i\right| = 1 \rightarrow C_p = C_q = 0, \ C_o = 1 \qquad (3.19)$$

Similarly,

$$D_c(\eta,\xi) = D_p\eta + D_q\xi + D_o \qquad (3.20)$$

where

$$\begin{aligned}
D_p &= k\left[(x_1 - x_3)u + (y_1 - y_3)v + (z_1 - z_3)w\right] \\
D_q &= k\left[(x_2 - x_3)u + (y_2 - y_3)v + (z_2 - z_3)w\right] \\
D_o &= k\left[x_3 u + y_3 v + z_3 w\right]
\end{aligned} \qquad (3.21)$$

16

Now the integral is given by

$$I_c = 2Ae^{jD_o} \left\{ e^{jD_p} \left[\frac{C_o}{D_p(D_q - D_p)} \right] - e^{jD_q} \left[\frac{C_o}{D_q(D_q - D_p)} \right] - \frac{C_o}{D_q D_p} \right\} \quad (3.22)$$

To avoid the numerical problems that occur near the singularities (denominators near zero), the following special cases are used:

Case 1: $|D_p| < L_t$ and $|D_q| \geq L_t$

$$I_c = \frac{2Ae^{jD_o}}{jD_q} \sum_{n=0}^{\infty} \frac{(jD_p)^n}{n!} \left\{ \frac{-C_o}{n+1} + e^{jD_q} C_o G(n,-D_q) \right\} \quad (3.23)$$

Case 2: $|D_p| < L_t$ and $|D_q| < L_t$

$$I_c = 2Ae^{jD_o} \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} \frac{C_o (jD_p)^n (jD_q)^m}{(m+n+2)!} \quad (3.24)$$

Case 3: $|D_p| \geq L_t$ and $|D_q| < L_t$

$$I_c = 2Ae^{jD_o} e^{jD_p} \sum_{n=0}^{\infty} \frac{(jD_q)^n}{n!} \left(\frac{C_o}{n+1} \right) G(n+1,-D_p) \quad (3.25)$$

Case 4: $|D_p| \geq L_t$, $|D_q| \geq L_t$, and $|D_q - D_p| < L_t$

$$I_c = \frac{2Ae^{jD_o}}{jD_q} \sum_{n=0}^{\infty} \frac{(jD_p - jD_q)^n}{n!} \left\{ -C_o G(n,D_q) + \frac{e^{jD_q} C_o}{n+1} \right\} \quad (3.26)$$

The function G is defined by

$$G(n,\gamma) = \int_0^1 s^n e^{j\gamma s} ds \quad (3.27)$$

There is recursive relationship that is efficient for finding G for higher values of n

$$G(n,\gamma) = \frac{e^{j\gamma} - nG(n-1,\gamma)}{j\gamma}, \quad n \geq 1 \quad (3.28)$$

17

and

$$G(0,\gamma) = \frac{e^{j\gamma} - 1}{j\gamma} \qquad (3.29)$$

C.   COORDINATE TRANSFORMATIONS

In general, the local facet coordinate system will not be aligned with the global coordinate system. A series of transformations must be performed in order to obtain the required relationships between the local and global coordinate variables and unit vectors. Figure 3 shows a facet with arbitrary orientation. The global coordinates are (x, y, z) and the local coordinates are (x'', y'', z''). Figure 4 shows that the z'' axis is aligned with the facet normal. Also, the figure shows a right triangle facet with the x'' and y'' axes aligned with the triangle edges, however, the transformation can be applied to any arbitrary facet that lies in the $x'' - y''$ plane.
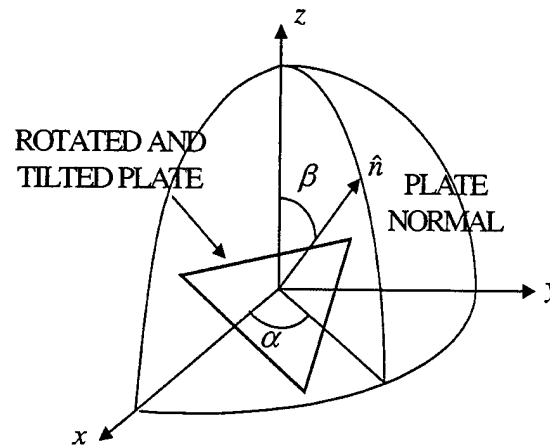
Figure 3. Facet with Arbitrary Orientation

18
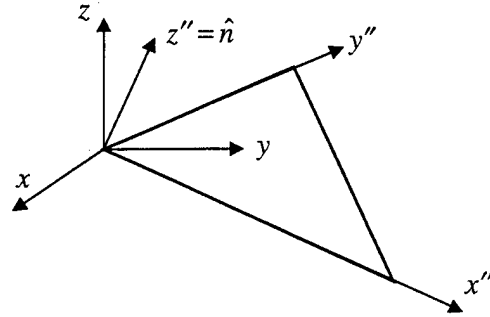
Figure 4. Facet Rotation Angles

The double-primed variables can be obtained by a series of two rotations of the global coordinate variables. The first is a rotation about the z axis by an angle α as shown in Figure 5. The resulting coordinate axes are referred to as (x', y', z'). The rotation angle is $\alpha = \text{atan2}(n_y, n_x)$ (atan2 is the MATLAB arctangent function that gives the proper quadrant). The second rotation is by an angle $\beta = \cos^{-1}(\hat{z} \bullet \hat{n})$ about the y' axis, illustrated in Figure 6.
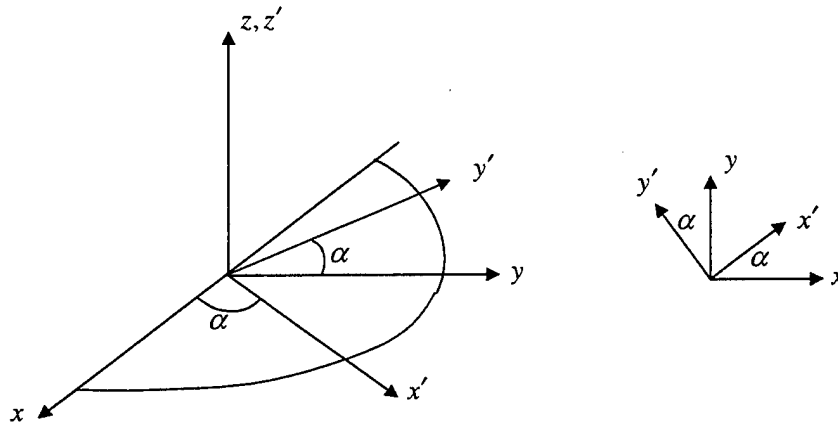
Figure 5. First Rotation by an Angle α about the z axis.

19
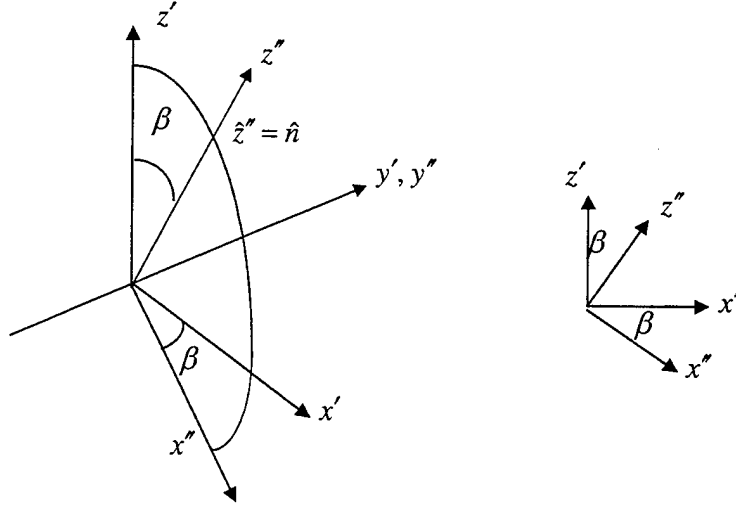
Figure 6. Second Rotation by an Angle β about the y' axis.

The transformation matrices are determined from geometry:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\overline{\overline{T'}}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad (3.30)$$

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix}}_{\overline{\overline{T''}}} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \qquad (3.31)$$

The double overbar denotes a matrix. Therefore,

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \overline{\overline{T''}}\,\overline{\overline{T'}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad (3.32)$$

The same transformation matrix holds for the unit vectors

$$\begin{bmatrix} \hat{x}'' \\ \hat{y}'' \\ \hat{z}'' \end{bmatrix} = \overline{\overline{T''}}\,\overline{\overline{T'}} \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} \qquad (3.33)$$

20

and the direction cosines

$$\begin{bmatrix} u'' \\ v'' \\ w'' \end{bmatrix} = \overline{\overline{T''}}\,\overline{\overline{T'}} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$ (3.34)

The direction cosines are

$$\begin{aligned} u &= \sin\theta\cos\phi \\ v &= \sin\theta\sin\phi \\ w &= \cos\theta \end{aligned}$$ (3.35)

and

$$\begin{aligned} u'' &= \sin\theta''\cos\phi'' \\ v'' &= \sin\theta''\sin\phi'' \\ w'' &= \cos\theta'' \end{aligned}$$ (3.36)

These quantities can be either incident angles (subscript "i") or observation angles (subscript "p"). For the monostatic case both are equal and subscripts are not required.

D. PHYSICAL OPTICS CURRENT

The components of the surface current in terms of the local cartesian coordinate variables are from [Ref. 6]

$$J_{x'} = \left[\frac{E_{i\theta'}\cos\theta''\cos\phi''}{2R_s + Z_o\cos\theta''} - \frac{E_{i\phi'}\cos\theta''\sin\phi''}{2R_s\cos\theta'' + Z_o} \right]$$ (3.37)

and

$$J_{y'} = \left[\frac{E_{i\theta'}\cos\theta''\sin\phi''}{2R_s + Z_o\cos\theta''} + \frac{E_{i\phi'}\cos\theta''\cos\phi''}{2R_s\cos\theta'' + Z_o} \right]$$ (3.38)

where R_s is the surface resistivity. When $R_s = 0$ ($|\Gamma| = 1$), the surface is a perfect electric conductor. As $R_s \to \infty$ the surface becomes transparent ($|\Gamma| \to 0$). Defining the transverse magnetic (TM) and transverse electric (TE) reflection coefficients as

21

$$\Gamma_{\text{TM}} = \Gamma_{\parallel} = \frac{-Z_o \cos\theta''}{2R_s + Z_o \cos\theta''} \tag{3.39}$$

$$\Gamma_{\text{TE}} = \Gamma_{\perp} = \frac{-Z_o}{2R_s \cos\theta'' + Z_o} \tag{3.40}$$

allows the surface current components to be expressed as

$$J_{x'} = \left[-\frac{E_{i\theta'} \cos\phi''}{Z_o} \Gamma_{\parallel} + \frac{E_{i\phi'} \sin\phi''}{Z_o} \Gamma_{\perp} \right] \cos\theta'' \tag{3.41}$$

$$J_{y'} = \left[-\frac{E_{i\theta'} \sin\phi''}{Z_o} \Gamma_{\parallel} - \frac{E_{i\phi'} \cos\phi''}{Z_o} \Gamma_{\perp} \right] \cos\theta'' \tag{3.42}$$

The spherical polar angles are known from the direction cosines

$$\begin{aligned} \sin\theta'' &= \sqrt{(u'')^2 + (v'')^2} \\ \cos\theta'' &= \sqrt{1 - (u'')^2 + (v'')^2} \\ \tan\phi'' &= \arctan\left(\frac{v''}{u''}\right) \end{aligned} \tag{3.43}$$

In the bistatic case, the incident angles are used in the above formulas.

E.   DIFFUSE CALCULATION

Models may have a rough surface due to irregularities or material imperfection. Surface roughness introduces path differences relative to a smooth surface and thus contribute a diffuse scattering component to the RCS that is added to the coherent component. The random deviation of the surface results in diffuse scattering which yields a relatively angle dependent term to the RCS. First we assume that the surface deviation resemble a Gaussian shape as shown in Figure 7. The extent of the Gaussian shape is characterized by a correlation distance C (in meters), which is the average distance at which deviations become uncorrelated. A large correlation distance implies a

slowly varying surface error, whereas a small correlation distance is associated with a rapidly varying error. The variance of the irregularities is denoted as δ^2 (or standard deviation as δ) as illustrated in Figure 8.

Rough Surface

Gaussian Approximation
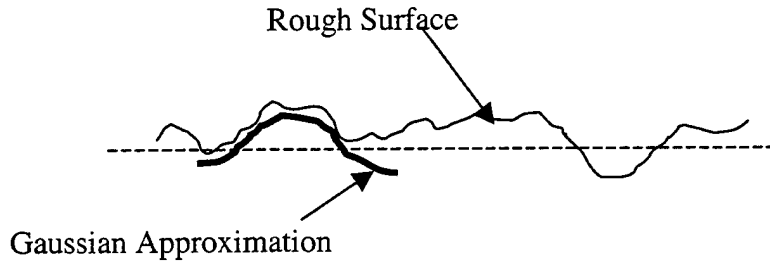
Figure 7. Gaussian Approximation of Rough Surface From [Ref. 1]

Changes slowly with x δ^2

(a)

Changes rapidly with x

(b)
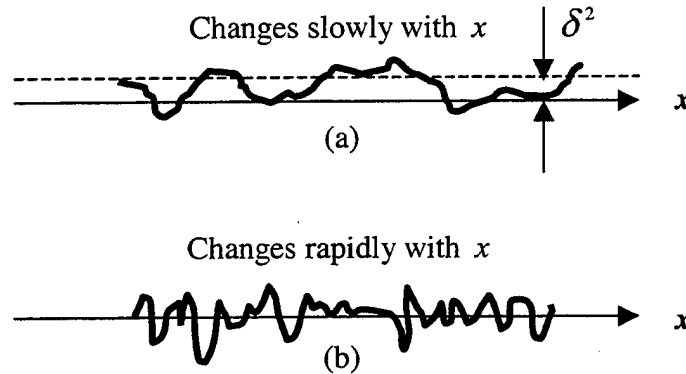
Figure 8. Correlation Distance. (a) Large Correlation Distance,
(b) Small Correlation Distance From [Ref. 1]

In the PO calculation, surface roughness is approximated by defining values for correlation distance and standard deviation. The average value of the RCS is approximately [Ref. 7]

$$\sigma_{ave} \approx \frac{4\pi A^2}{\lambda^2}\cos^2\theta \, e^{-4k^2\delta^2}\left[\frac{4k^2\delta^2\pi C^2}{A}\exp\left\{-\frac{C^2\pi^2\sin^2\theta}{\lambda^2}\right\}\right] \qquad (3.44)$$

23

where A is the area of the facet and θ is measured from the normal. This value is simply added to the specular term

$$\sigma_{total} = \sigma_{po} + \sigma_{ave} \qquad (3.45)$$

F.   THE SCATTERED FIELD OF A COLLECTION OF FACETS

An arbitrary target can be approximated by a collection of triangular facets. Equation 3.16 must be evaluated for all facets, and the polarization factors and coordinate systems may differ for each facet. However, the integral over all facets will have the form of equation 3.17. To obtain the scattered field of the array of facets the following procedure is used:

1. The incident field is specified in the global coordinates: angle of incidence (θ_i, ϕ_i) and polarization $(E_{i\theta}, E_{i\phi})$.

2.   For each facet (indexed from $c = 1, 2, ..., N_c$) the incident field is transformed to the local facet coordinate system. Also the angle of incidence is determined in the local facet coordinate system.

3. The physical optics current is computed using the incident magnetic field, and the radiation integral evaluated over the facet surface to obtain the scattered field components $(E_{s\theta''}, E_{s\phi''})$.

4. The scattered field components are transformed back to the global coordinate system, thereby obtaining $(E_{s\theta}, E_{s\phi})$ for each facet.

5.   All of the N_c transformed scattered field components are superimposed (vector summed) to obtain the total scattered field in the observation direction.

6. The RCS is computed from the total scattered field from equation (2.3).

A flow diagram of the procedure for the monostatic calculation is shown in Figure 9. For a bistatic calculation, the current does not have to be recomputed for each observation angle because the incident angle, and hence current, is fixed. [Ref. 8]
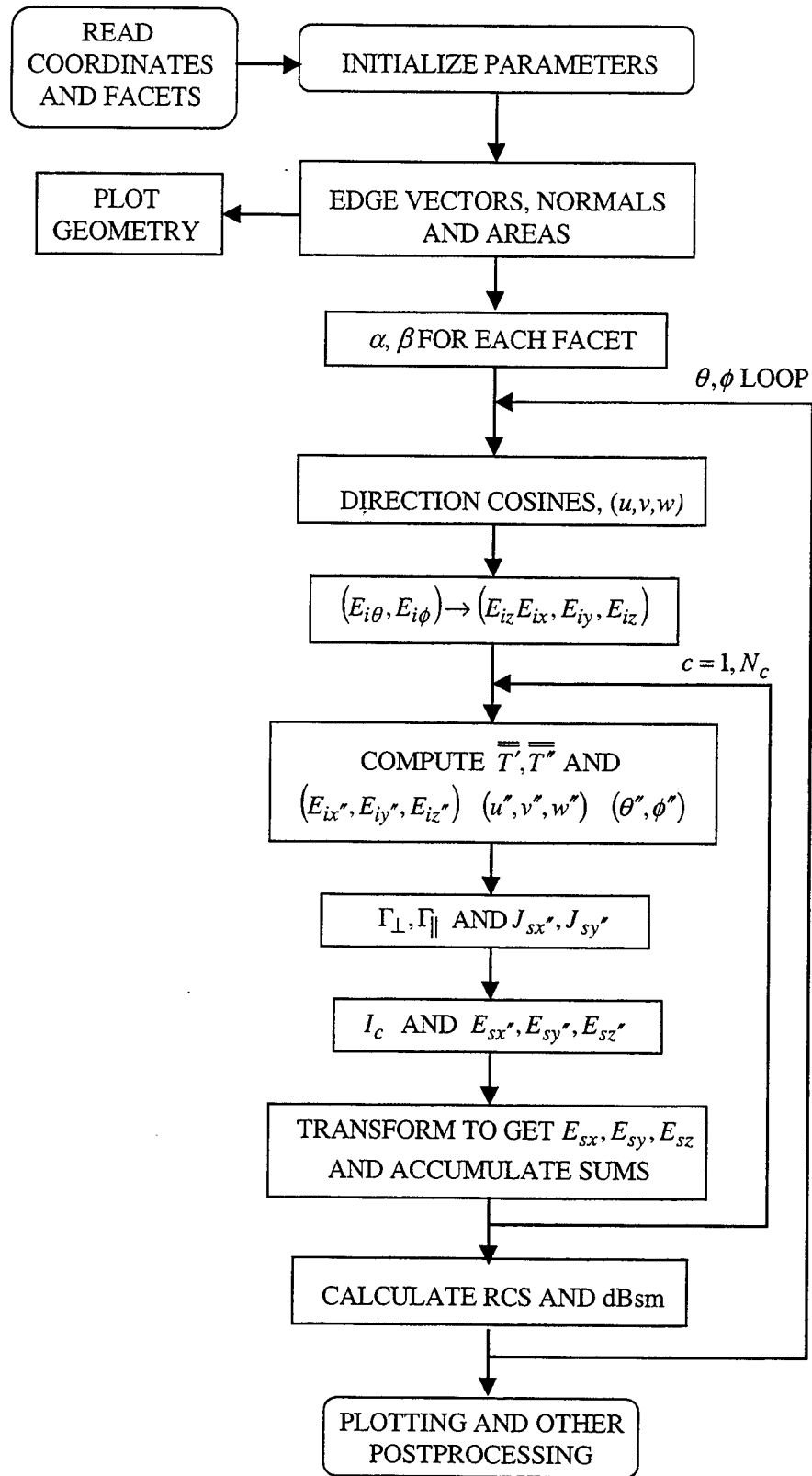
Figure 9. Flow Chart for Computing Monostatic RCS of an Array of Facets

IV.   THE POFACETS GRAPHICAL USER INTERFACE

POFACETS is a graphical user interface (GUI) radar cross section (RCS) prediction code based on the physical optics (PO) approximation and implemented in MATLAB.   It is accurate when the target is large in terms of wavelength.   Scattering objects are approximated by arrays of triangles (facets) and superposition is used to compute the total RCS of the model object as described in Chapter III.   Multiple reflections are not included.   The GUI consists of three modules: (1) *Design Model*, (2) *Calculate Monostatic RCS*, and (3) *Calculate Bistatic RCS*.

Design Model provides a convenient way of modeling a complex object or loading and editing a previously designed model.   The geometry of the model is defined using two sets of data. The first set defines the x, y, z coordinates of the vertices.   The second set of data defines the component triangular facets of the object, their nodes, and illumination and resistivity characteristics.

Calculate Monostatic RCS computes the total radar cross section of the target using a monostatic radar (co-located transmitter and receiver).   The scattered field of each triangle is computed as if it was isolated and other triangles were not present.   Multiple reflections, edge diffraction and surface waves are not included.   Shadowing is approximately included by considering a facet to be completely illuminated or completely shadowed by the incident wave.   Options for defining the roughness of the surface are available.

Calculate Bistatic RCS computes the total radar cross section of the target using a bistatic radar (transmitter and receiver at different locations). The incidence direction is fixed and the observation direction changes.

To run the POFACETS GUI, the user must first run MATLAB and set the path or directory to where the POFACETS code is stored. This may be done in two ways. The first is by using the *File Menu* and selecting *Set Path*. This displays the Path Browser window and from here click on the *Browse* button. In the Browse for Folder window, select the folder where POFACETS is located then click the *OK* button. The second method is by using the change directory command at the MATLAB command prompt to where the code is stored such as *cd c:\pofacets*. After setting the path or directory, typing *pofacets* at the command line will execute the GUI as shown in Figure 10. A brief explanation of the POFACETS GUI may be obtained if the user clicks on the *Help* button, which displays the POFACETS Help GUI as shown in Figure 11.
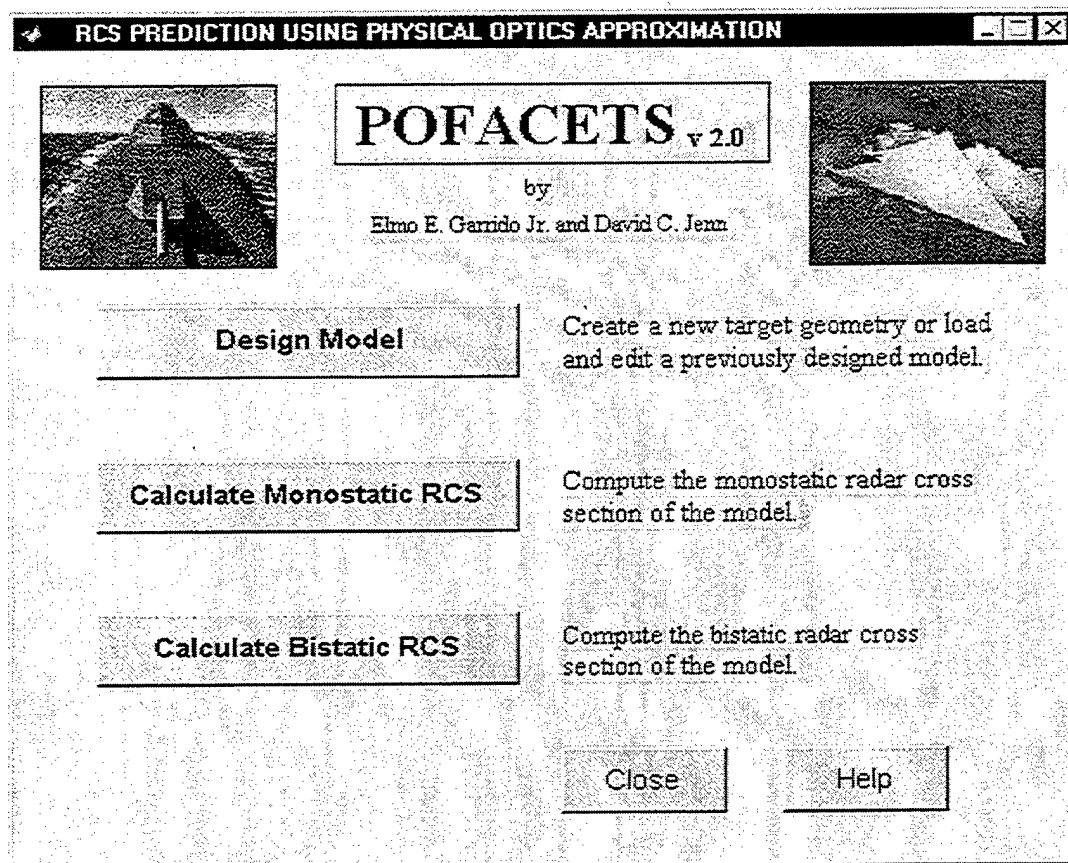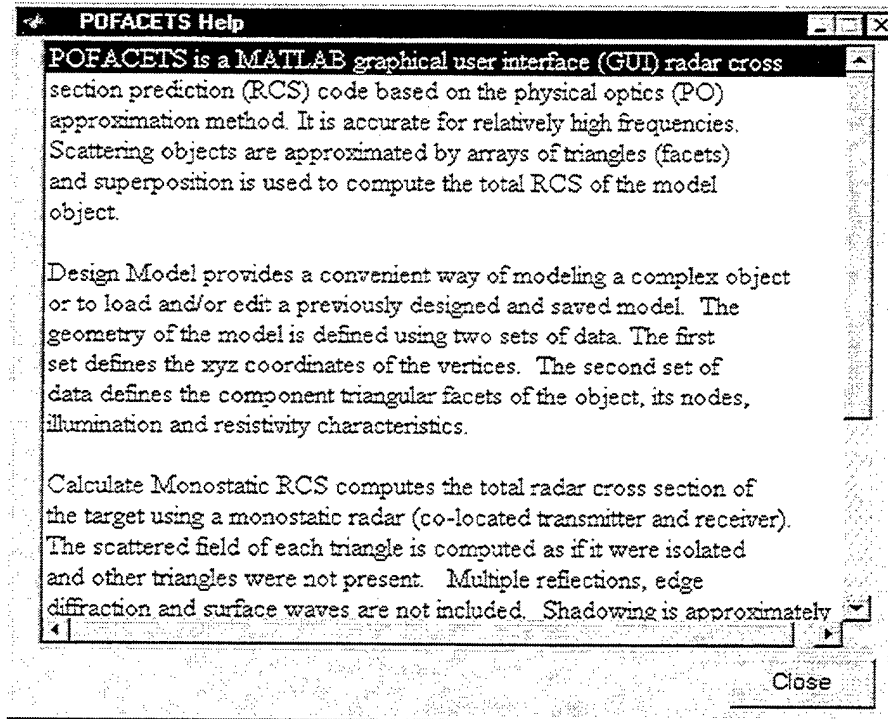
Figure 10. POFACETS Main GUI

Figure 11. POFACETS Help GUI

A. DESIGNING AND PLOTTING A MODEL

To illustrate the *Design Model* module of the GUI, we will design the AIM-9

Sidewinder missile shown below in Figure 12 by modeling it with the set of coordinates
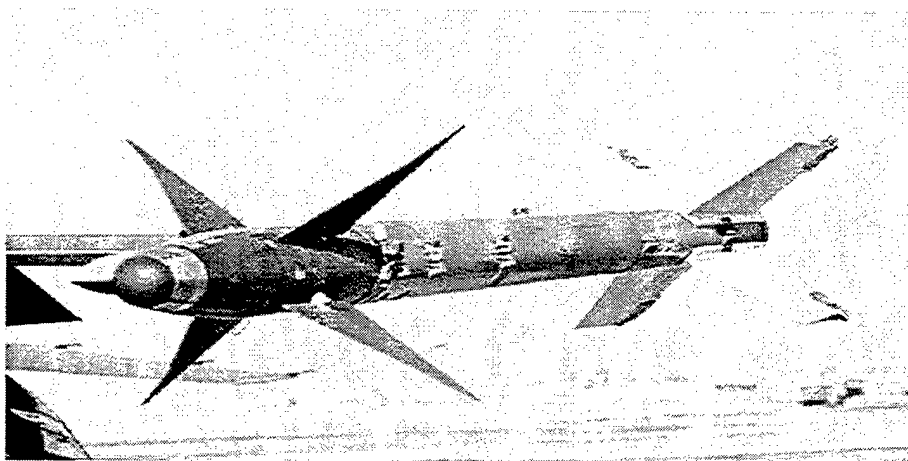
in Table 1 and facet configuration in Table 2.

Figure 12. AIM-9 Sidewinder from [Ref. 9]

30

Vertex #	X (m)	Y (m)	Z (m)	Vertex #	X (m)	Y (m)	Z (m)
1	0.000	0.000	0.00	20	-0.250	0.065	0.150
2	0.130	0.000	0.00	21	-0.250	0.065	0.000
3	0.130	0.130	0.00	22	0.065	0.000	0.000
4	0.000	0.130	0.00	23	0.065	0.000	0.200
5	0.000	0.000	2.57	24	0.065	-0.250	0.150
6	0.130	0.000	2.57	25	0.065	-0.250	0.000
7	0.130	0.130	2.57	26	0.130	0.065	2.200
8	0.000	0.130	2.57	27	0.130	0.065	2.400
9	0.065	0.065	2.87	28	0.380	0.065	2.200
10	0.130	0.065	0.00	29	0.065	0.130	2.200
11	0.130	0.065	020	30	0.130	0.065	2.400
12	0.380	0.065	015	31	0.065	0.380	2.200
13	0.380	0.065	0.00	32	0.000	0.065	2.200
14	0.065	0.130	0.000	33	0.000	0.065	2.400
15	0.065	0.130	0.200	34	-0.380	0.065	2.200
16	0.065	0.385	0.150	35	0.065	0.000	2.200
17	0.065	0.385	0.000	36	0.065	0.000	2.400
18	0.000	0.065	0.000	37	0.065	-0.380	2.200
19	0.000	0.065	0.200				

Table 1. AIM-9 Sidewinder Coordinate Set

Facet #	Node 1	Node 2	Node 3	Illumination	Normalized Resistivity
1	3	2	1	1	1
2	1	4	3	1	1
3	3	7	6	1	1
4	6	2	3	1	1
5	4	8	7	1	1
6	7	3	4	1	1
7	4	1	5	1	1
8	5	8	4	1	1
9	2	6	5	1	1
10	5	1	2	1	1
11	2	9	6	1	1
12	8	9	7	1	1
13	5	9	8	1	1
14	5	6	9	1	1
15	10	11	12	0	1
16	12	13	10	0	1
17	16	15	14	0	1
18	14	17	16	0	1
19	20	19	18	0	1
20	18	21	20	0	1
21	22	23	24	0	1
22	24	25	22	0	1
23	26	27	28	0	1
24	29	31	30	0	1
25	32	34	33	0	1
26	35	36	37	0	1

Table 2. AIM-9 Sidewinder Facet Configuration

To design a new object, the user clicks on the *Design Model* button of Figure 10. This displays the Design Model Coordinates GUI shown in Figure 13.
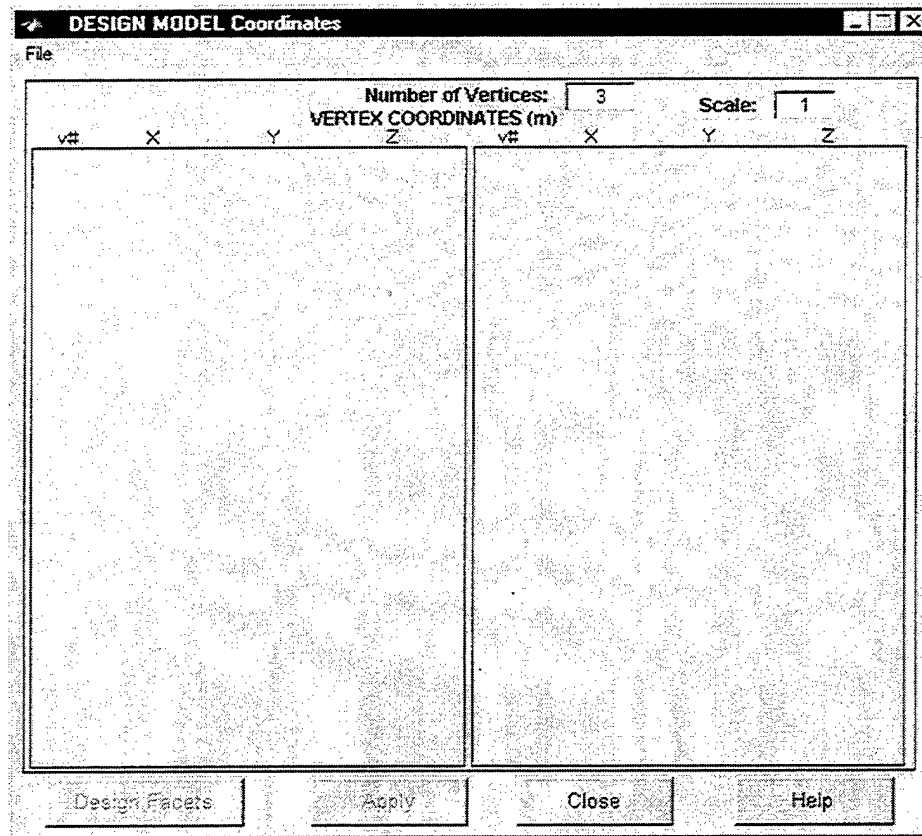

Figure 13. Design Model Coordinates GUI

The *Number of Vertices* is initially disabled but selecting *File | New* in the menu will enable this box to accept an entry. Since the Sidewinder example has 37 vertices, we enter this number in the appropriate box and the GUI generates the corresponding number of editable text boxes (all initialized to zero) to accommodate the Cartesian coordinates of the model as well as enable the *Apply* button to check all the individual coordinates later. Entering the coordinates correctly and clicking on the *Apply* button

33

displays the Coordinate Status GUI (Figure 14) and after clicking the *OK* button, enables

the *Design Facets* button and the active GUI should now look like Figure 15.
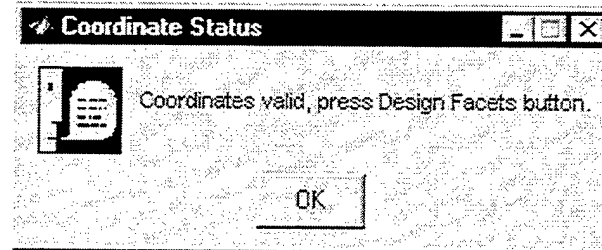
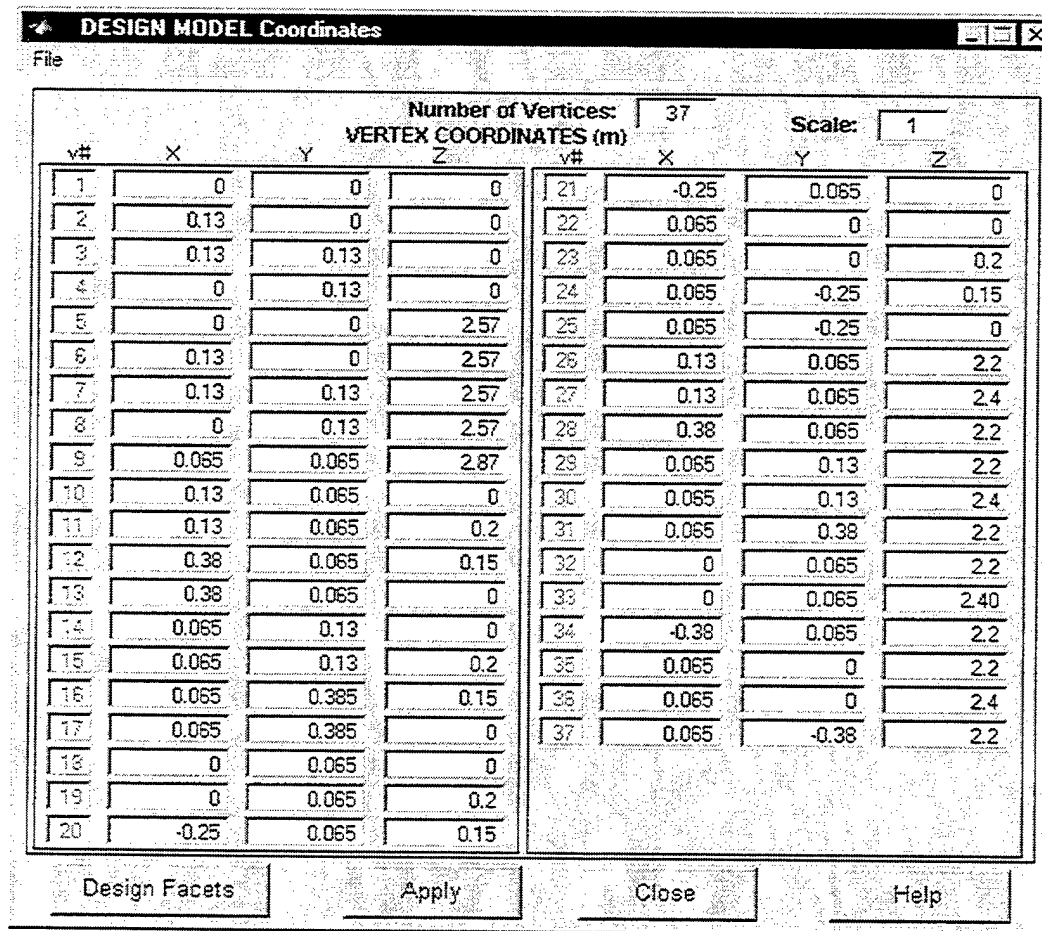

Figure 14. Coordinate Status GUI

Figure 15. Design Model Coordinates GUI with Vertices of the Sidewinder

Clicking the *Design Facets* button of Figure 15 will display the Design Model Facets GUI as shown in Figure 16, which is ready to accept the number of facets parameter.
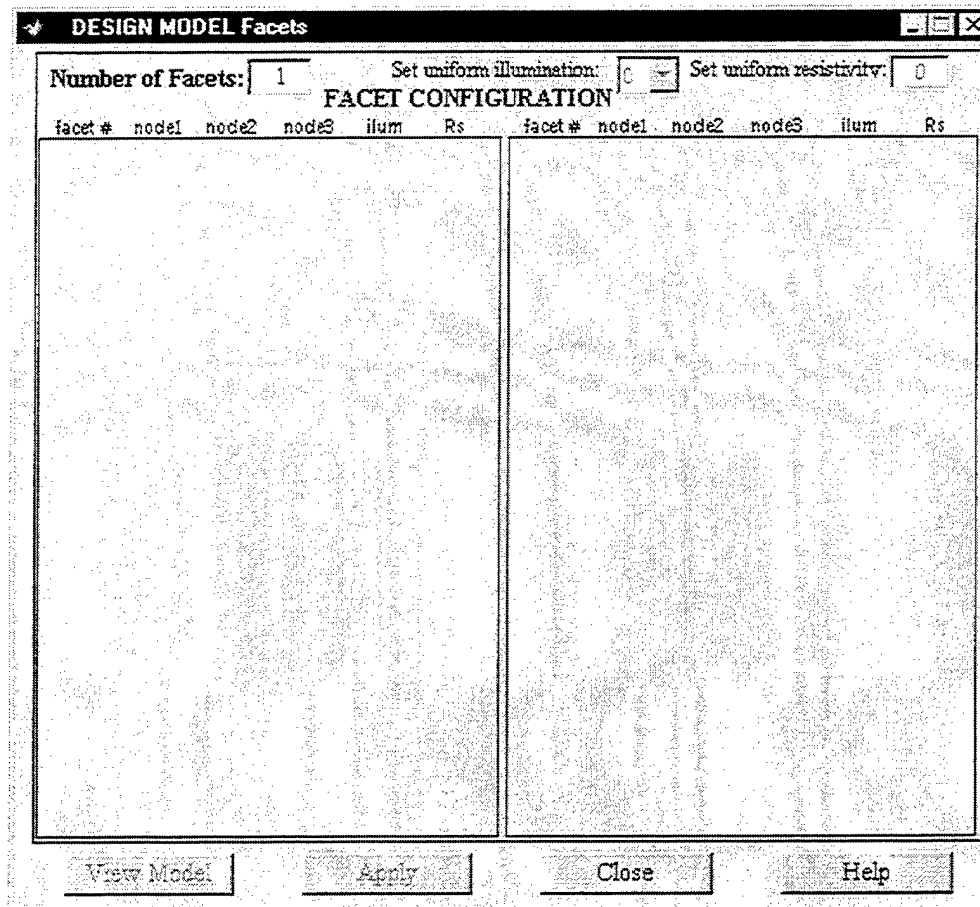

Figure 16. Design Model Facets GUI

The *Set uniform illumination* and *Set uniform resistivity* uicontrols are initially disabled. Inputting a value for the number of facets, in this case 26 (based on Table 2), enables these functions to be set by the user. The GUI generates the corresponding number of boxes for the user to define the nodes, illumination and resistivity characteristics for each of the sequenced component facet number, and the *Apply* button

35

is enabled. The nodes and illumination values are then encoded using the data in Table 2. The illumination parameter specifies whether both sides of a facet are illuminated by the incident wave. If *ilum* = 0 both sides of the plate are illuminated (e.g., a fin or a wing); if *ilum* = 1 only the outer face is illuminated (e.g., a fuselage). The normalized resistivity $R_s / 377$ is used to define the resistivity characteristics of the model surface. For example, to conveniently define the entire missile skin to have an impedance of 377 ohms, *uniform resistivity* is set to 1 and it will be noted that the resistivity values for all the facets are accordingly set in the *Rs* column. Pressing the *Apply* button displays the Facet Configuration Status GUI (Figure 17) if there are no errors and subsequently clicking the *OK* button enables the *View Model* button and the active GUI should now look like Figure 18.
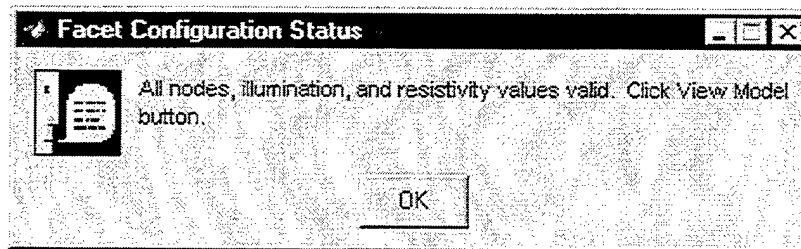

Figure 17. Facet Configuration Status GUI

To visualize the model comprised of its triangular components, the plot may be seen by clicking the *View Model* button, which displays the Model Geometry GUI as shown in Figure 19.
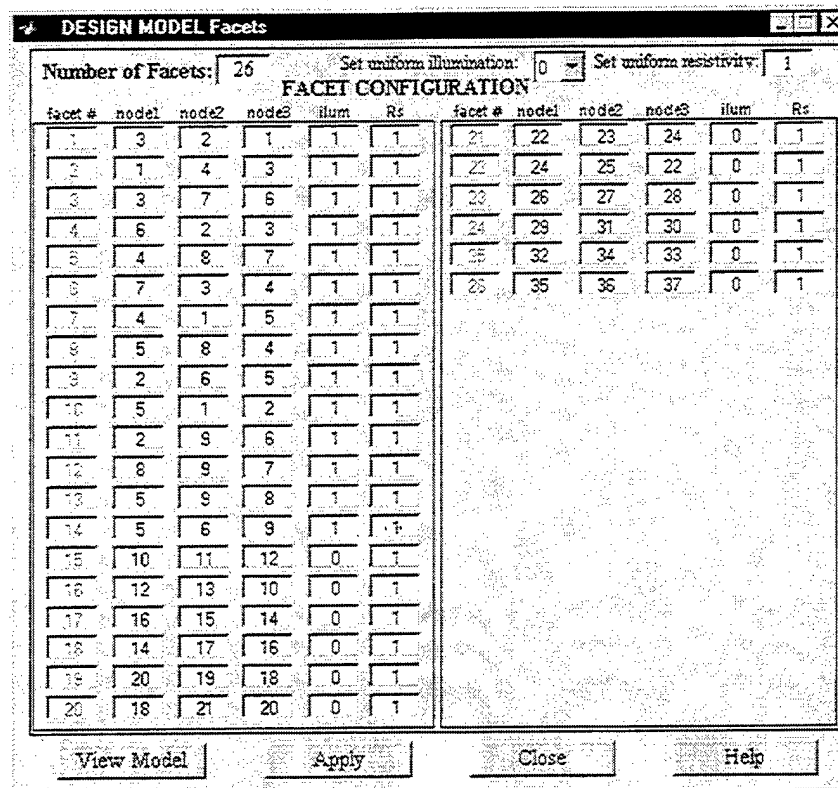
36

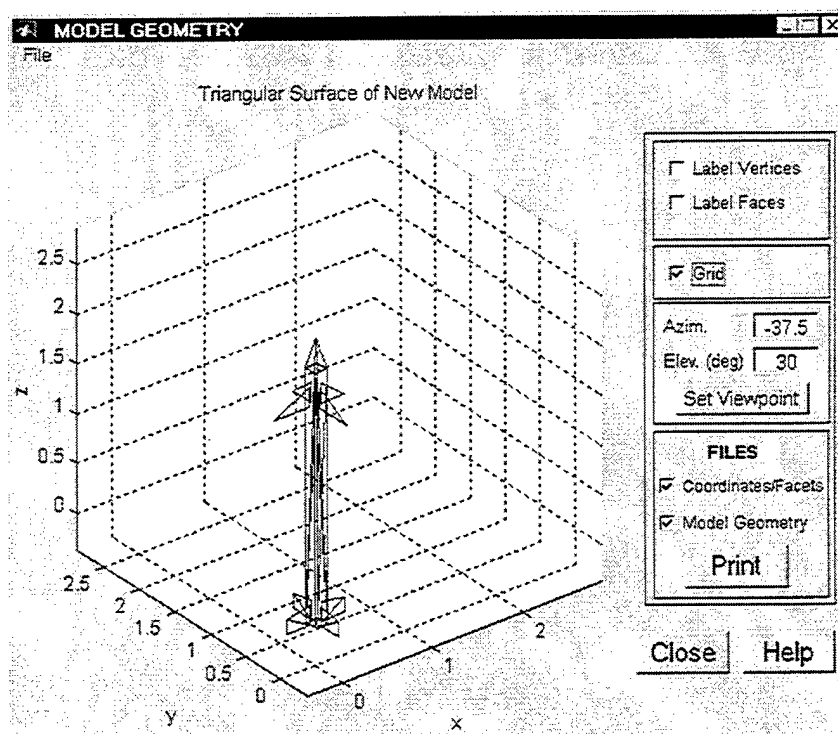Figure 18. Design Model Facets GUI with Facet Configuration of the Sidewinder

Figure 19. Model Geometry GUI

37

1. Help GUIs

A number of *Help GUIs* are available to provide the user valuable assistance in working with the *Design Model* module. The three *Help GUIs* may be viewed by clicking on the *Help* buttons of the Design Model Coordinates, Design Model Facets, and the Model Geometry GUIs to display Figures 20, 21 and 22, respectively.
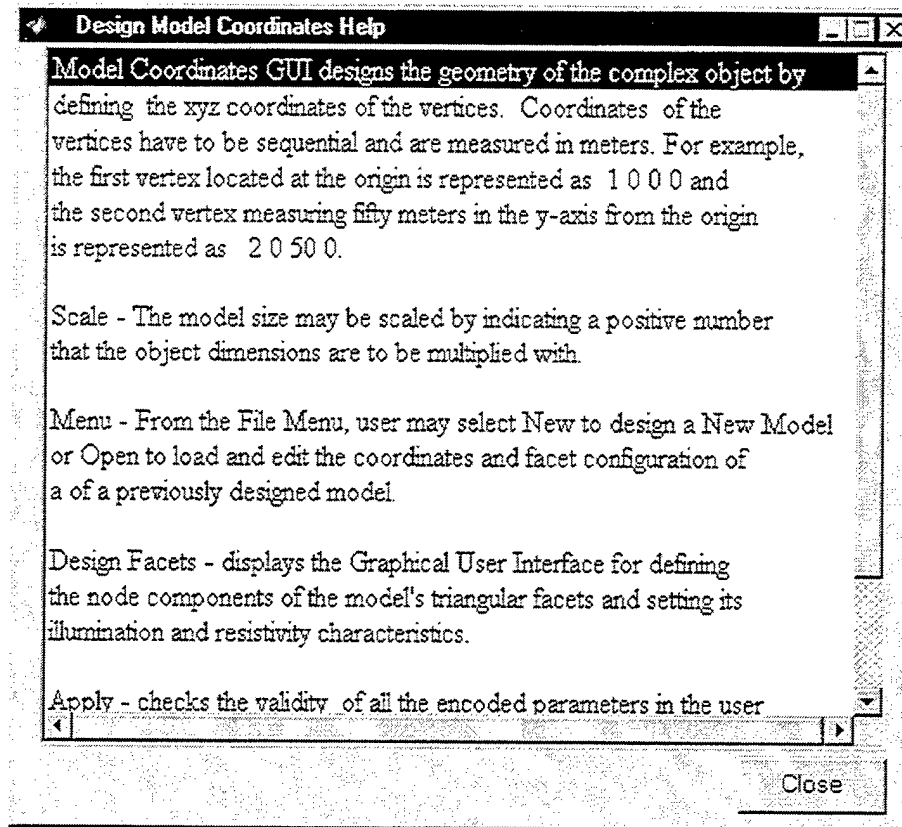

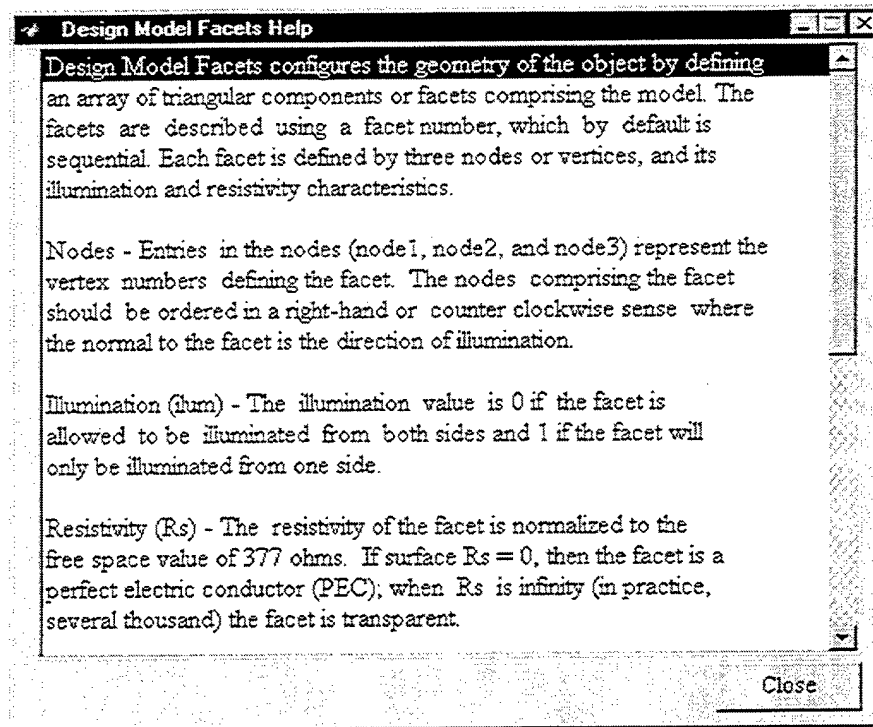Figure 20. Design Model Coordinates Help GUI

```
Design Model Facets Help                                          [_][□][X]

Design Model Facets configures the geometry of the object by defining
an array of triangular components or facets comprising the model. The
facets are described using a facet number, which by default is
sequential. Each facet is defined by three nodes or vertices, and its
illumination and resistivity characteristics.

Nodes - Entries in the nodes (node1, node2, and node3) represent the
vertex numbers defining the facet. The nodes comprising the facet
should be ordered in a right-hand or counter clockwise sense where
the normal to the facet is the direction of illumination.

Illumination (ilum) - The illumination value is 0 if the facet is
allowed to be illuminated from both sides and 1 if the facet will
only be illuminated from one side.

Resistivity (Rs) - The resistivity of the facet is normalized to the
free space value of 377 ohms. If surface Rs = 0, then the facet is a
perfect electric conductor (PEC); when Rs is infinity (in practice,
several thousand) the facet is transparent.

                                                              Close
```

Figure 21. Design Model Facets Help GUI

```
Model Geometry Help                                              [_][□][X]

The plot of the Model Geometry is a scaled version of the design
defined by the coordinates and the facets configuration.

Label Vertices - is a function that plots the model geometry showing
the vertex numbers as defined in the Design Target Model-Coordinates
GUI

Label Faces - is a function that plots the model geometry showing the
facet or triangle number as defined by the nodes in the Design Target
Model - Facets.

Grid On/Off - is a function that plots or clears the axis area of the
model geometry figure of grid lines to enhance visualization of the
model.

Print Files - this function prints a hard copy of the coordinates/facets
data and/or the model geometry as selected.

Close - closes the GUI figure.

                                                              Close
```
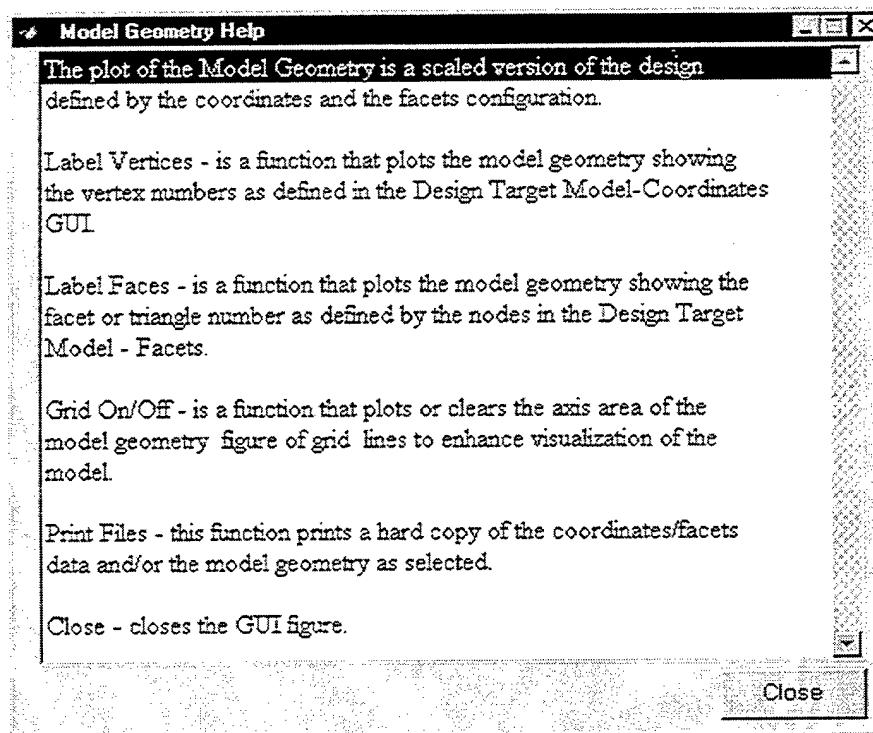
Figure 22. Model Geometry Help GUI

2. Status GUIs

Displaying several *Status GUIs* have been incorporated in the code to draw user attention to specific part of the GUI requiring action. For the Design Model Coordinates GUI, an error in the *Number of Vertices* entered will display the status shown in Figure 23. Likewise, since the Design Model Coordinates GUI can initially accommodate a maximum of 40 vertices only, exceeding this limit will display Figure 24. Meanwhile, encoding an invalid *Scale* value will show Figure 25.
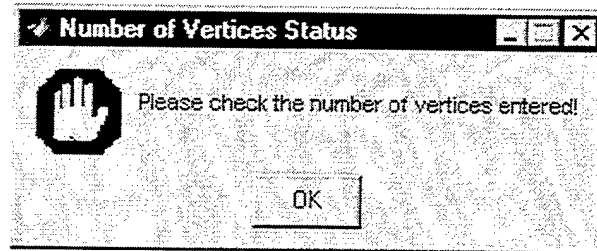


Figure 23. Number of Vertices Status GUI
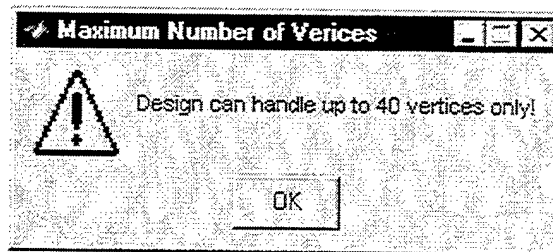
Figure 24. Maximum Number of Vertices Status GUI
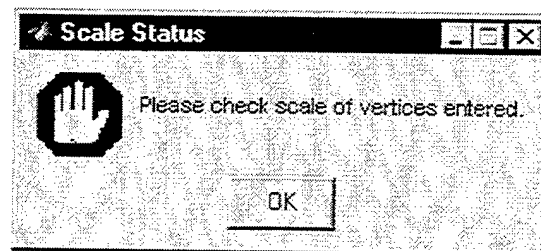
Figure 25. Scale Status GUI

40

The user is advised of erroneous entries on the coordinates with the display of Figures 26a to c, while a valid set of coordinates is ascertained by Figure 26d.
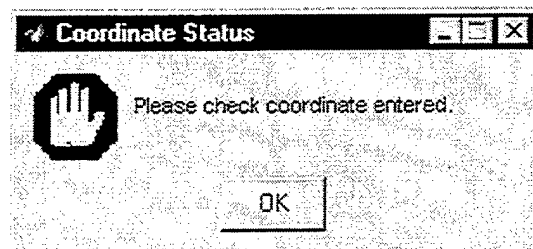
(a)
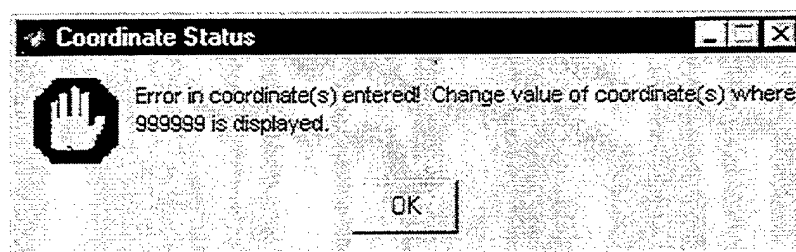
(b)
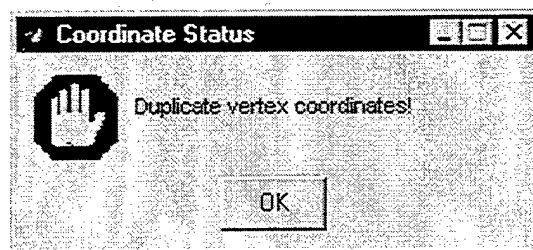
(c)
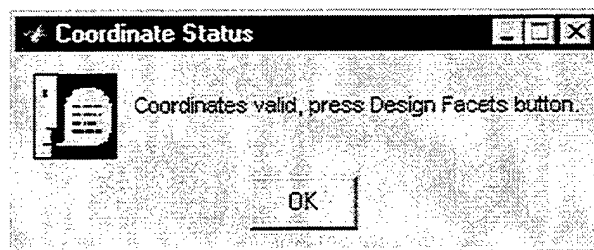
(d)

Figure 26. Coordinate Status GUIs. (a) Invalid Coordinate, (b) Uncorrected Erroneous Coordinate, (c) Duplicate Set of Coordinates, (d) Valid Set of Coordinates

41

For the Design Model Facets GUI, an error in the *Number of Facets* entered will display Figure 27. Likewise, since the GUI can initially accommodate a maximum of 40 facets only, exceeding this limit will display Figure 28. Meanwhile, encoding an invalid *resistivity* value for a particular facet will show Figure 29.
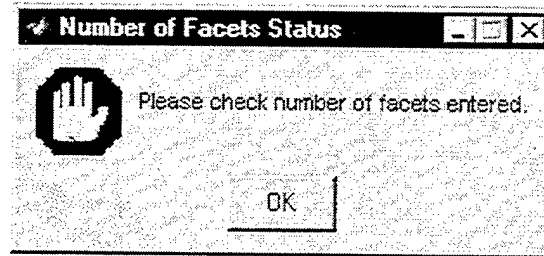
Figure 27. Number of Facets Status GUI
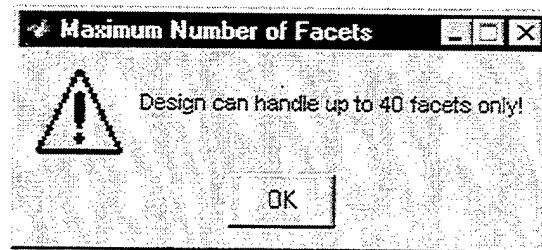
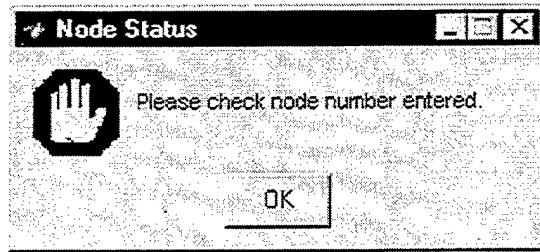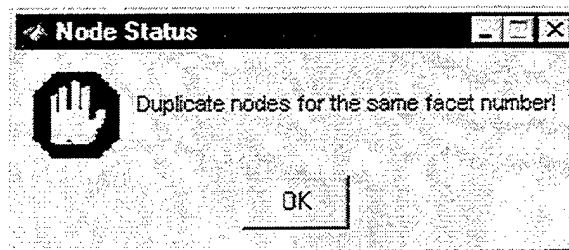
Figure 28. Maximum Number of Facets GUI
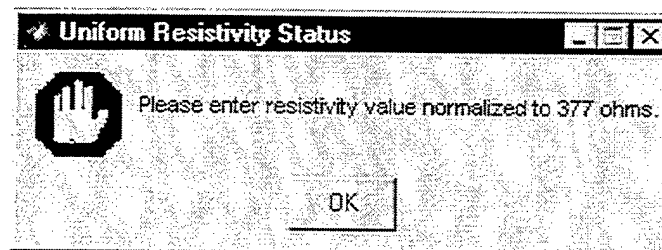
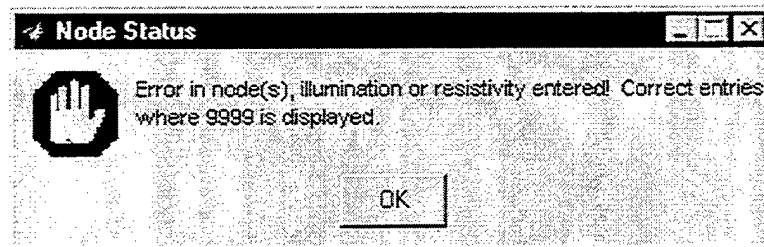Figure 29. Uniform Resistivity Status GUI

The user is advised of erroneous entries in the configuration of the facet(s) with messages shown in Figures 30a to f and encoding valid parameters is ascertained by Figure 30g.
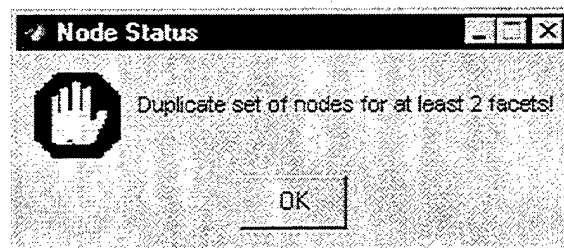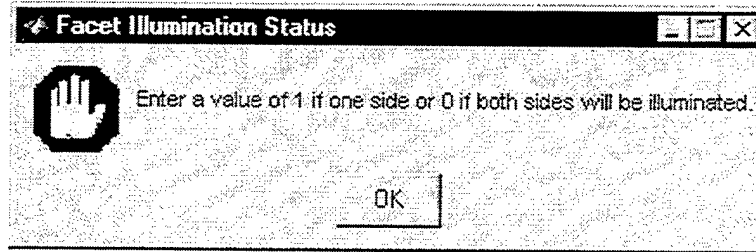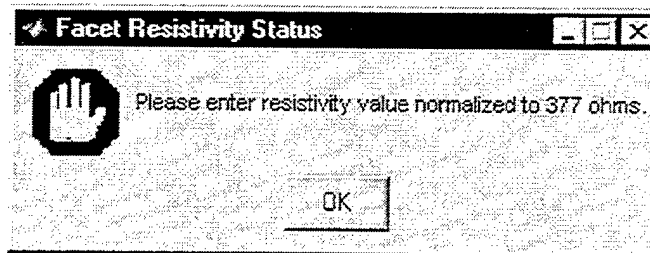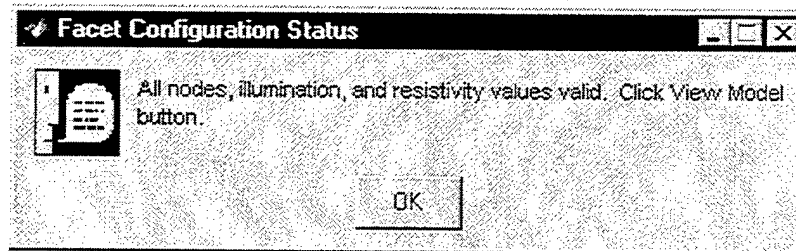
42

(a)

(b)

(c)

(d)

Figure 30. Facet Configuration Status GUIs. (a) Invalid Node, (b) Duplicate Nodes,
(c) Uncorrected Erroneous Node, (d) Duplicate Set of Nodes

(e)

(f)

(g)

Figure 30 (continued). Facet Configuration Status GUIs. (e) Invalid Illumination Value, (f) Invalid Facet Resistivity, (g) Valid Facet Configuration

3. Model Geometry GUI Functionalities

The Model Geometry GUI shown in Figure 19 provides some functionalities to enhance user visualization of the model object and to verify correctness of node and facet designation. These include plotting the vertex/facet numbers with the object or the grid lines by selecting any or all of the following checkboxes: *Label Vertices, Label Faces,* and *Grid,* respectively. Selecting the *Coordinates/Facets* and *Model Geometry* checkboxes and clicking on the *Print* button will generate a hard copy of the coordinates and facets data.

44

4. Saving the Model Files

The model's file is composed of two sets of data, coordinate set and facet configuration. However, to save a model only the filename is required. To save the Sidewinder example, click on *File | Save As ...* from the menu and this displays the New File GUI shown in Figure 31. We enter the filename *Sidewinder* and click on the *Save* button. The program saves the files and confirms the action to the user by displaying the File Saved GUI shown in Figure 32. Inadvertently specifying a filename that already exists in the list of models will prompt the user to overwrite said file with the Replace File GUI shown in Figure 33.
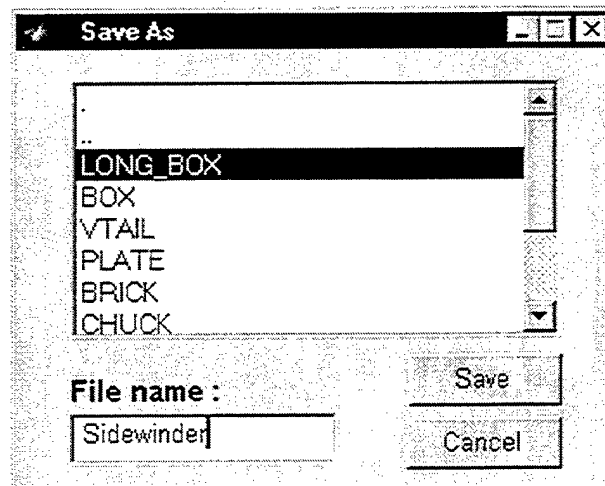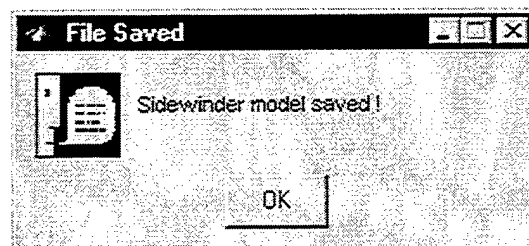

Figure 31. New File GUI
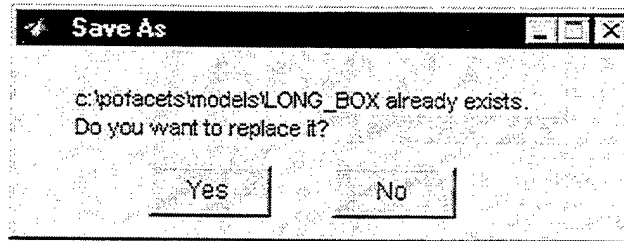

Figure 32. File Saved GUI

45

Figure 33. Replace File GUI

5. Editing an Existing Model

To edit an existing model, its set of data needs to be retrieved and displayed before editing or scaling can proceed. This is accomplished by clicking on the *File | Open* menu of the Design Model Coordinates GUI (Figure 10) which displays the Open Model File GUI shown in Figure 34. Suppose we would like to edit the VTAIL model. We select it among the listed file names and click the *Open* button. This action performs a series of operations that will display its data set and the model plot as shown in Figure 35.
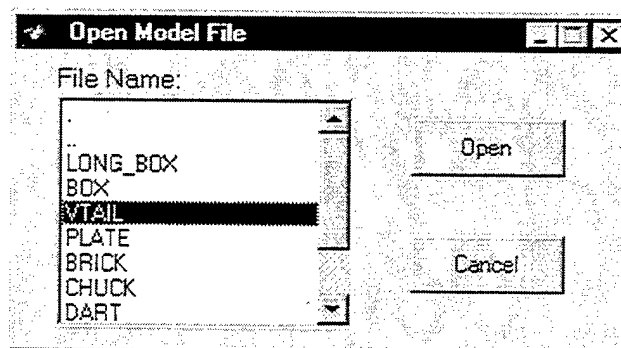

Figure 34. Open Model File GUI

Figure 35. Displayed VTAIL File

At this point, the user may click on any of the three overlapping GUIs to make it active and edit the pertinent data. The different *Status GUIs* mentioned earlier will guide the user throughout the use of the interface.

B. CALCULATING THE MONOSTATIC RCS

1. Smooth PEC Model

To compute the Monostatic RCS of a model object, the user clicks on the *Calculate Monostatic RCS* button of the POFACETS Main GUI (Figure 10). This clears the screen of other GUIs (except the Main GUI) and brings up the Calculate Monostatic RCS GUI with default values and parameters shown in Figure 36.

Figure 36. Calculate Monostatic RCS GUI

For this case, we will calculate the monostatic RCS of a closed box model with a smooth surface, measuring 100 meters on each side and which is made of perfect electric conductor (PEC) material. Since this is a closed form object, only the outer face will be illuminated and all the facets will have a uniform illumination of 1. We further assume that the incident wave is θ-polarized (TM-z), the frequency of interest 15 MHz and we would like the RCS measured at $\phi = 0^0$ plane while varying θ from 0 to 360 degrees to get a view of the model RCS around its sides.

Setting up the GUI with the parameters mentioned should result in the *Calculate Monostatic RCS GUI* that looks like Figure 37 and the RCS plot shown in Figure 38.

48

Two computational parameters that may be varied are the *Taylor Series Region* and the *Number of Terms* in the Taylor Series. See [Ref. 5] for details on how these parameters affect the accuracy and computation time. In general, the default values give good results with relatively fast run times. Also, a two-dimensional contour can be generated by setting θ values from 0 to 90, the ϕ values from 0 to 360, increment angle to 2 degrees, and the resulting computation of the monostatic RCS is shown in Figure 39.

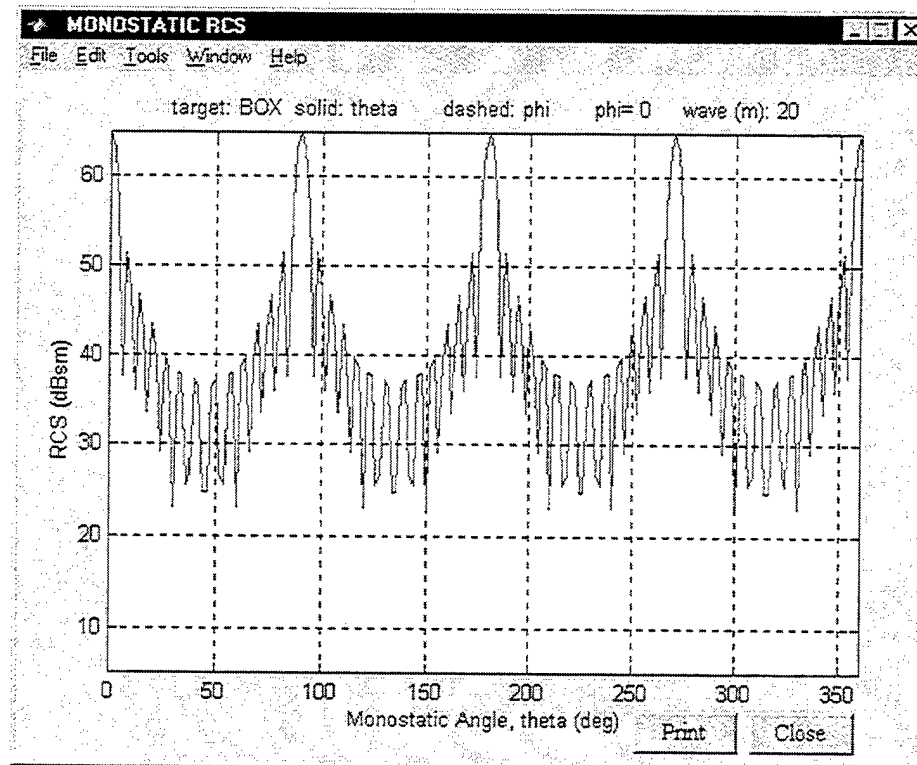Figure 37. Calculate Monostatic RCS GUI for Smooth PEC Box Model

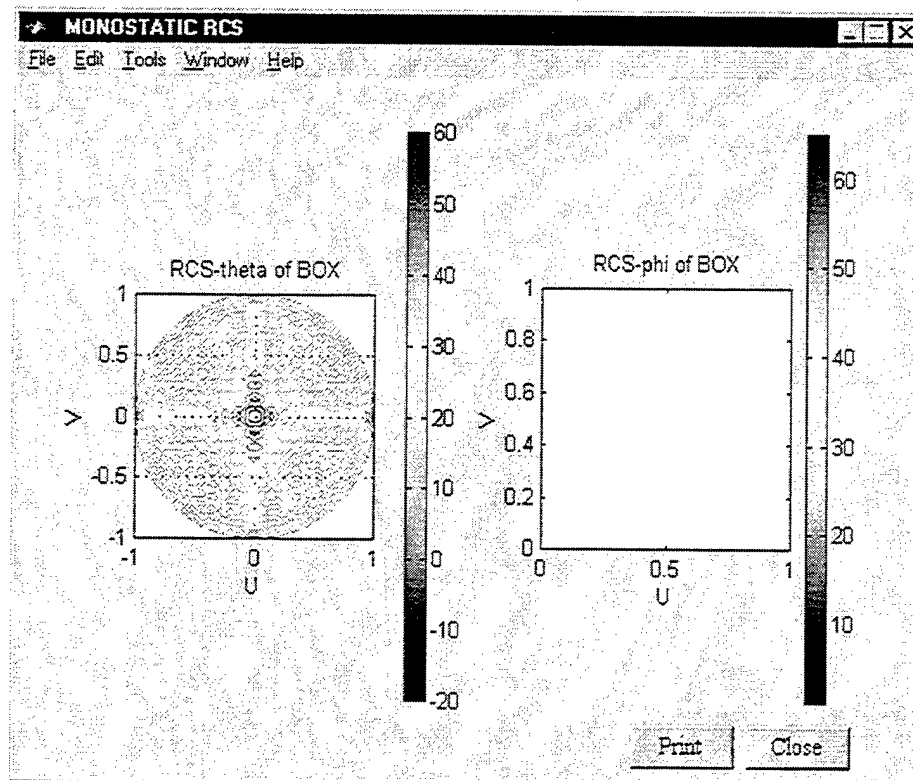Figure 38. Monostatic RCS of Smooth PEC Box Model

Figure 39. Contour Plot of Smooth PEC Box Model

50

RCS-theta is specified as $\sigma_{\theta\theta}$ (θ polarized incident wave and θ polarized receiver). This is the co-polarized component. RCS-phi is $\sigma_{\phi\theta}$ (θ polarized incident wave and ϕ polarized receiver). The latter is the cross-polarized component. In this particular example, it happens to be at a level below the specific threshold in the code.

2.   Model with Surface Impedance

An object coated with a radar absorbing material or having a skin made of resistive material may be approximated by defining its surface to have an impedance value. Equations 3.37 and 3.38 describes the components of the surface current at the target which is inversely proportional to the resistivity of the surface. Using the box model as an example and defining its surface to have an impedance three times the value of free space (i.e. uniform resistivity is set to 3), we see that the monostatic RCS calculation (Figure 40) is 15dB down in the specular return and from 15 to 20dB at the other various angles compared to the PEC model shown in Figure 38.

3.   Rough Surface Model

Meanwhile, to observe the effect of the diffused scattering component of a rough surface, the box model is defined to have a correlation distance of 100 meters (or equivalently 5λ) and a standard deviation about the mean value of 2 meters (that is, 0.1λ for a frequency of 15 MHz). Without varying the other parameters, the monostatic RCS plot in Figure 41 shows a decline of about 8 dB in the specular direction and an increase in the RCS at all the other angles due to the diffuse scattering effect compared to the PEC model results in Figure 38.
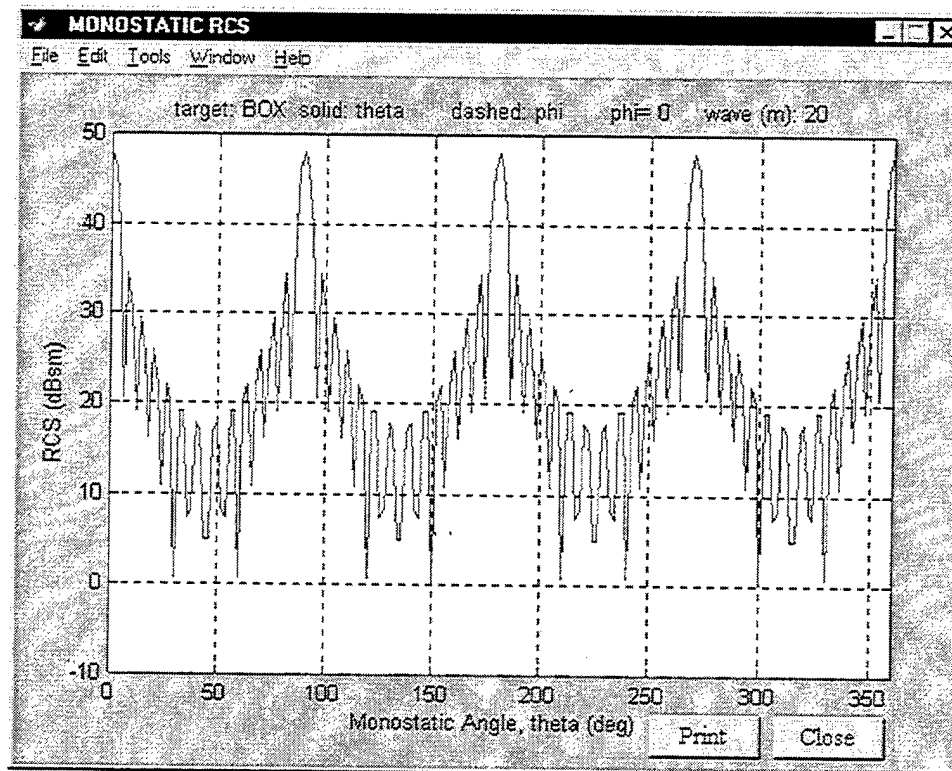
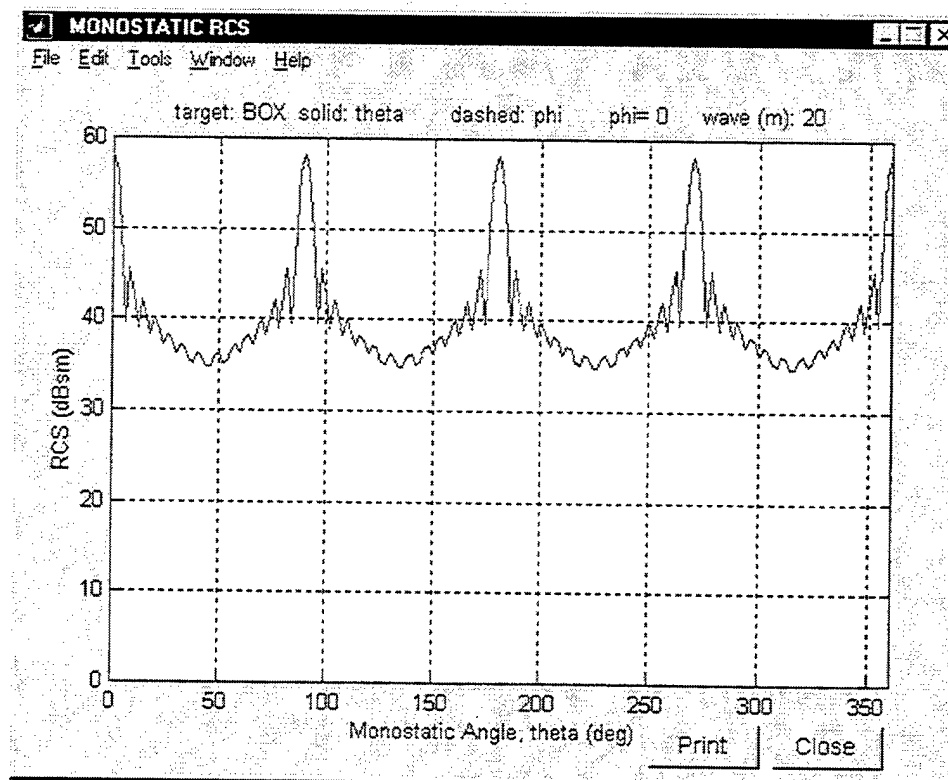Figure 40. Monostatic RCS of Smooth Box Model with Surface Impedance

Figure 41. Monostatic RCS of Box Model with Rough Surface
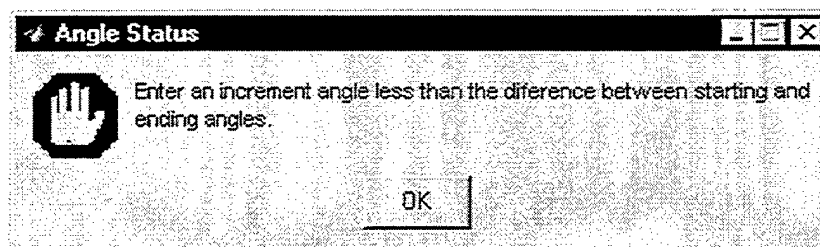
4. Calculate Monostatic RCS GUIs

Several GUIs were designed to enable the user to pinpoint specific parameters requiring attention. Errors in encoding values for the theta and phi angles display Figures 42a to c, while setting the same starting and ending angles advise the user that a theta or phi-cut is set as shown in Figure 43.
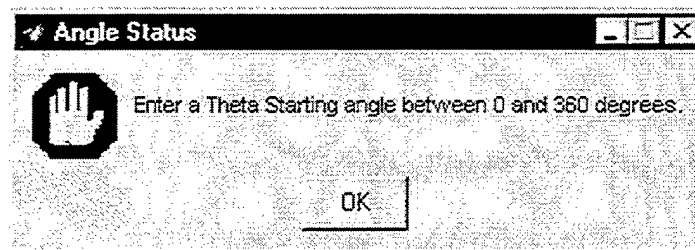
(a)
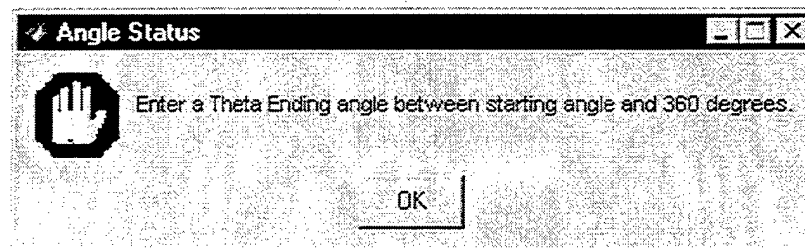
(b)

(c)

Figure 42. Angle Status GUIs. (a) Invalid Theta Starting Angle, (b) Invalid Theta Ending Angle, (c) Invalid Theta Increment Angle
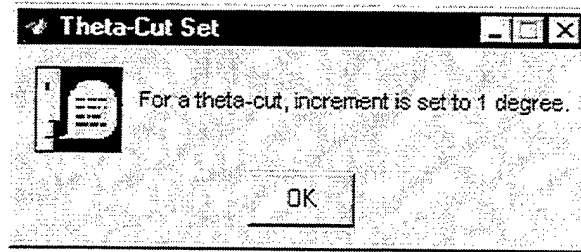
Figure 43. Theta-Cut Set GUI

Other GUIs include notifying the user of the need to correct entries for the *Frequency*, *Wavelength*, and the *Length of the Taylor Series Region* shown in Figures 44, 45, and 46, respectively.
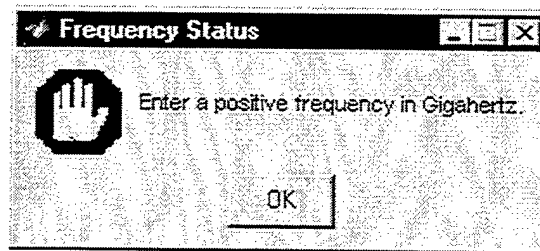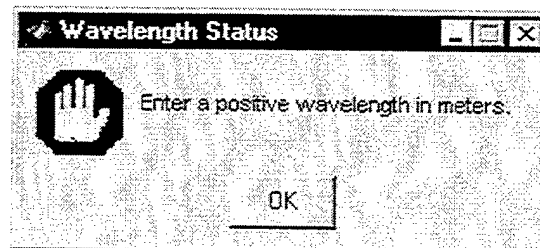

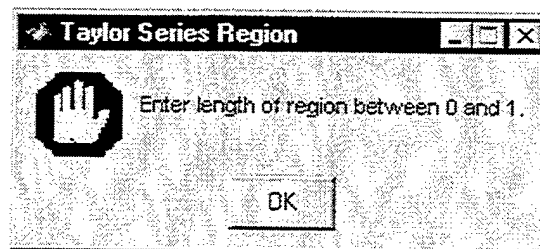Figure 44. Frequency Status GUI

Figure 45. Wavelength Status GUI

Figure 46. Taylor Series Region GUI

54

In addition, some GUIs have been pre-defined to warn the user if the parameter entered will take some time to generate the results as in Figure 47 or suggest a range of values such as Figures 48 and 49.
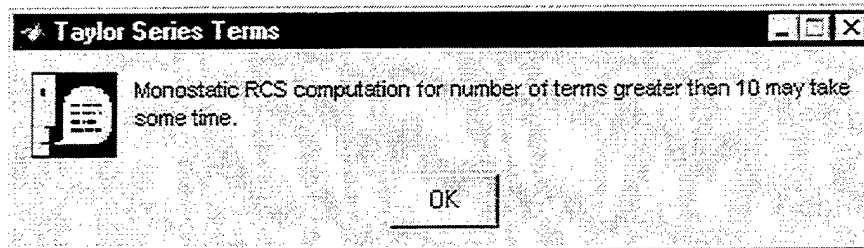
Figure 47. Taylor Series Terms GUI
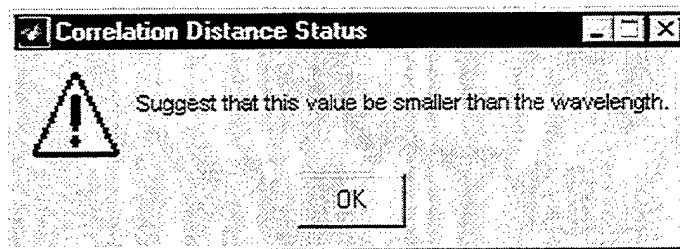
Figure 48. Correlation Distance Status GUI
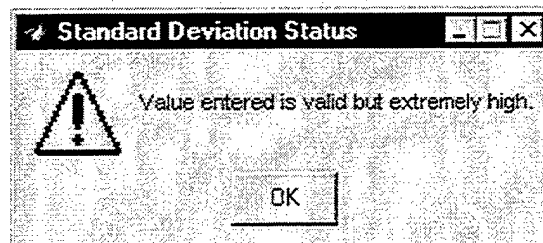
Figure 49. Standard Deviation Status GUI

It will be evident to the user that the *Calculate RCS* button is initially disabled. Selecting a model from the list box and clicking the *Load File* button will confirm to the user the model selected and give a reminder to set all the parameters as shown in Figure 50. Clicking the *OK* button will enable the *Calculate RCS* button. Finally, clicking the *Calculate RCS* button informs the user that the GUI is calculating the monostatic RCS of

the model selected as shown in Figure 51 below. The *Calculate Monostatic RCS Help*

GUI shown in Figure 52 can also provide some valuable information.
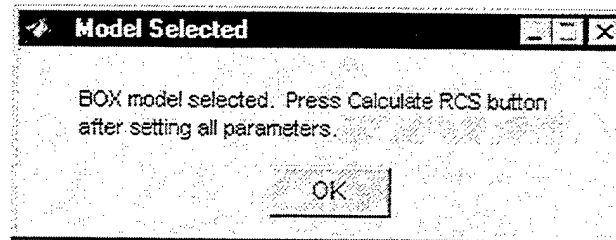
Figure 50. Model Selected GUI
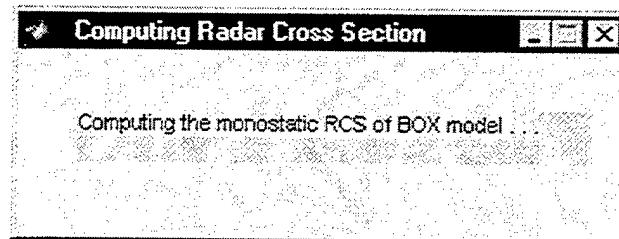
Figure 51. Computing Monostatic RCS GUI
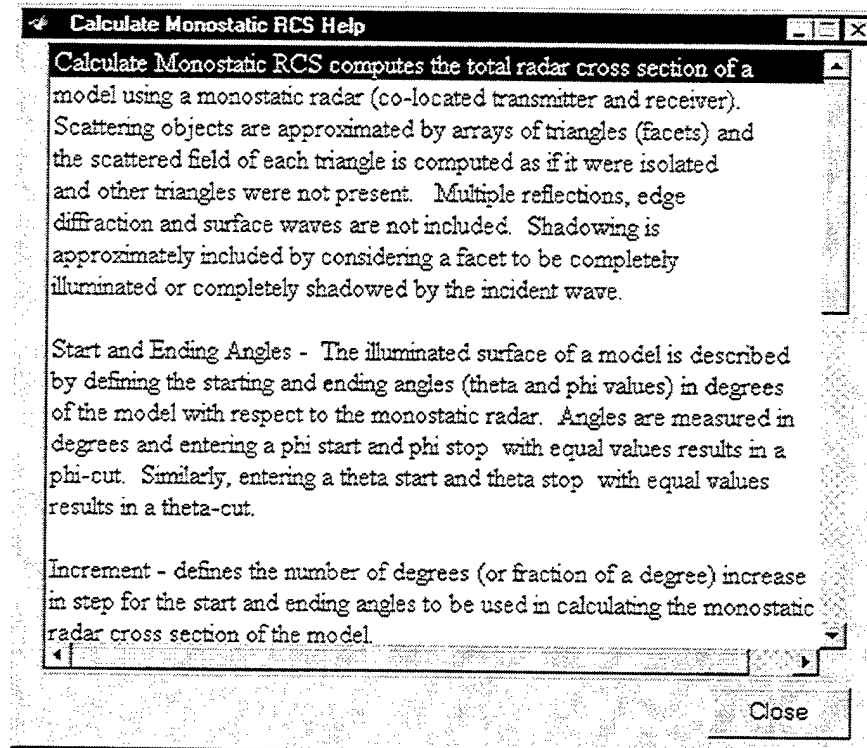
Figure 52. Calculate Monostatic RCS Help GUI

56

C.   CALCULATING THE BISTATIC RCS

The *Calculate Bistatic RCS GUI* (Figure 53) is very similar to the *Calculate Monostatic RCS GUI* (Figure 36) except for the θ and ϕ incident angles that define the transmitter location with respect to the object.

Figure 53. Calculate Bistatic RCS GUI

For an incidence angle of 45 degrees in the $\phi = 0^{0}$ plane and the observation direction varied from 0 to 360 degrees in the $\phi = 0^{0}$ plane, using a frequency of 0.2 GHz and the number of Taylor Series terms set at 3, the bistatic RCS of the box model is shown in Figure 54. The lobe at 135 degrees is the specular reflection from the top. The forward scatter is also evident. Note that the forward scatter is higher than the highest

57

monostatic backscatter lobe. The accompanying ray diagram in Figure 53 illustrates the

directions of the expected large RCS lobes.
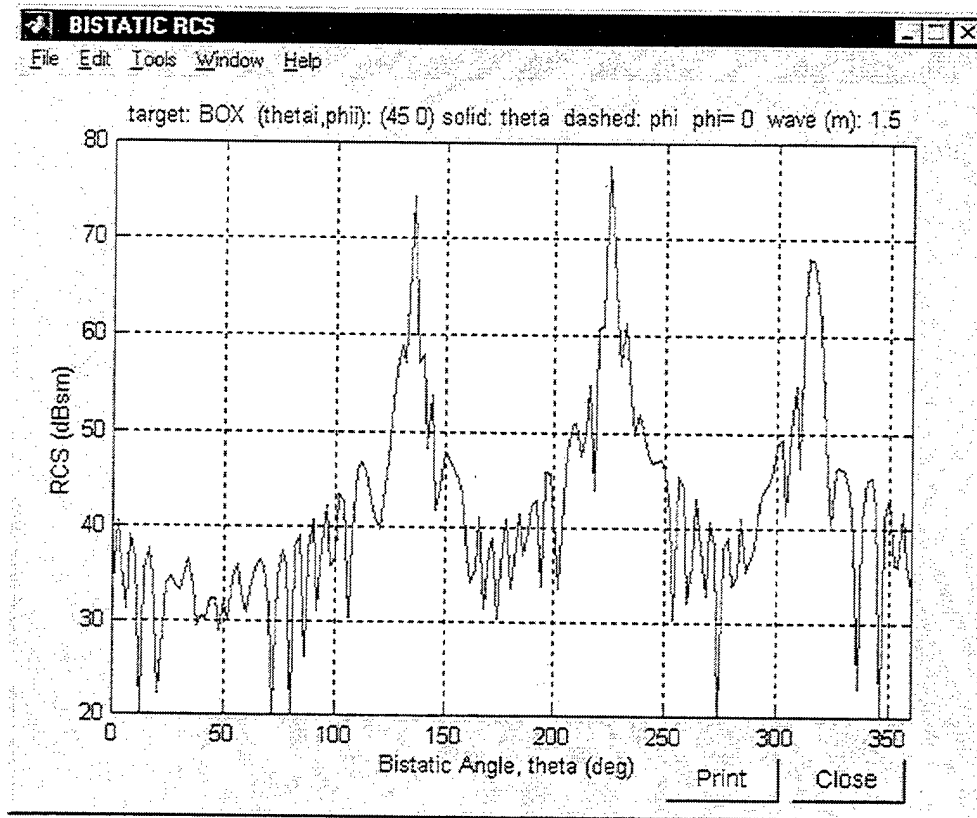
Figure 54. Bistatic RCS of Box Model
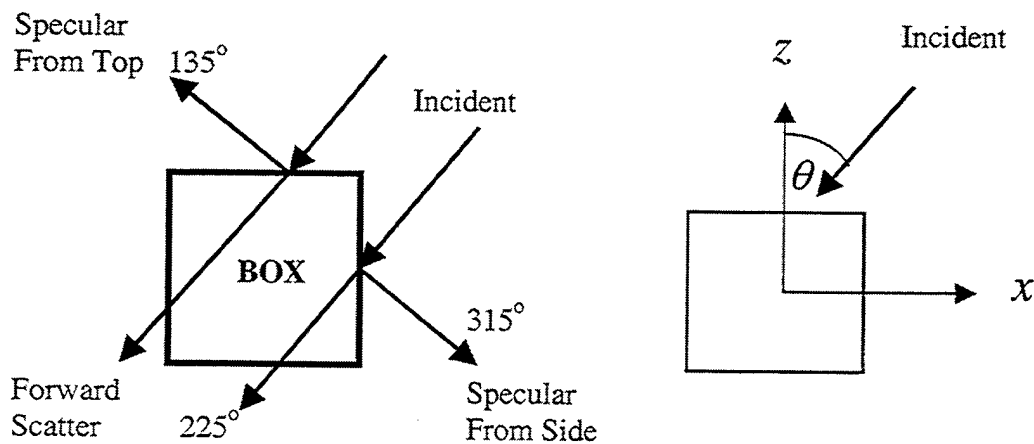
Figure 55. Ray Diagram for the Bistatic RCS of Box Model. (a) Forward and Specular
Scatter, (b) Incident Polarization

Status GUIs similar to that of the *Calculate Monostatic RCS* module were designed to enable the user to easily correct specific parameters. *The Calculate Bistatic RCS*, however has an additional GUI for the incidence angle as shown in Figure 56. Other helpful tips may be found in *Calculate Bistatic RCS Help GUI* shown in Figure 57.
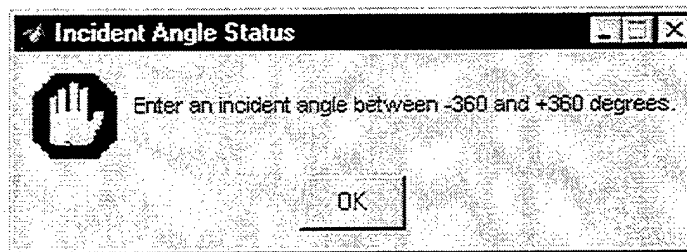
Figure 56. Incident Angle Status GUI
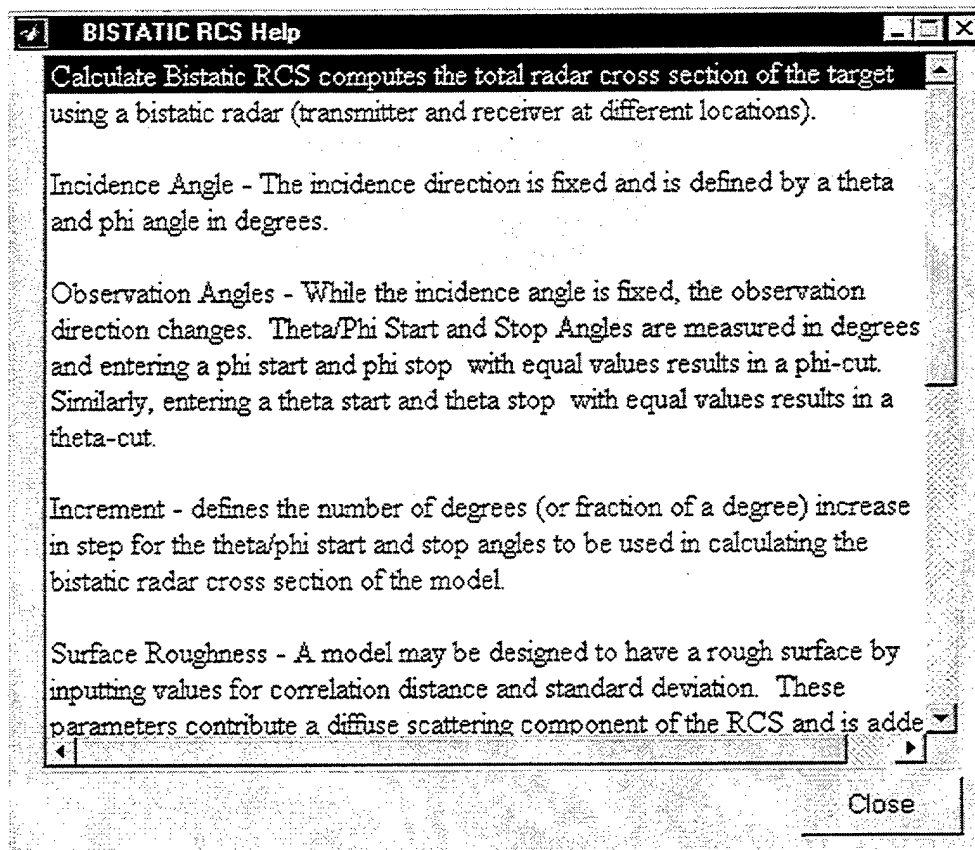
Figure 57. Calculate Bistatic RCS Help GUI

THIS PAGE INTENTIONALLY LEFT BLANK

V. SUMMARY AND RECOMMENDATIONS

A. SUMMARY

The main purpose of this work was to develop a Graphical User Interface (GUI) that will provide a convenient way of modeling complex objects using triangular facets, provide options to define the surface roughness of the scattering object, and calculate the monostatic or bistatic radar cross section over a range of frequencies or wavelengths of interest. Care was taken to design the GUI to make it appealing and easy to use and several validation techniques were incorporated to make it robust and prevent the user from encoding erroneous entries. Help windows and messages guide the user in making efficient use of the GUI. The GUI was implemented in MATLAB utilizing its scientific computing features and several built-in plotting functions to visualize the model geometry and results of the RCS calculation. Implementing the code as a user interface is the best way to avoid errors that may have significant consequences related to the use of this tool for the desired purpose.

The GUI can be used by both students and electromagnetic professionals to quickly determine the RCS performance of their model by varying the computational parameters to suit their design requirements.

B. RECOMMENDATIONS

The GUI developed was based on the basic physical optics approximation. An important area for future work is to be able to upgrade this GUI into one that incorporates other scattering mechanisms, primarily the effects of multiple reflections. Multiple reflections complicate the PO approximation because the incident field on a patch is not only determined by the direct field from the source, but also the field scattered by other

61

facets in the direction of the triangle under consideration. Numerical integrations will be required that substantially increase the computation time.

APPENDIX A. MATLAB M-FILES

The following are the description of the Matlab script files contained in the POFACETS Graphical User Interface:

1.   BPlot.m

This file generates either a theta of phi-cut bistatic RCS calculation plot of a selected model.

2.   BPlot1.m

This file generates a contour plot of the bistatic RCS calculation of a selected model.

3.   BMessage.m

This file displays the name of the model selected in the Calculate Bistatic RCS GUI and enables the Calculate RCS button.

4.   G.m

This file calculates the recursive Green's function using the two parameters (n and γ) passed.

5.   MPlot.m

This file generates either a theta of phi-cut monostatic RCS calculation plot of a selected model.

6.   MPlot1.m

This file generates a contour plot of the monostatic RCS calculation of a selected model.

7. MaxPoints.m

This file displays a message that the GUI design can accommodate only a maximum of 40 vertices only.

8. Repl_File.m

This file displays a GUI prompting the user to replace the data of an existing model.

9. Save_As.m

This file displays a GUI with a list of the existing models and prompts the user to enter a filename.

10. SvAs.m

This file implements the functionality of the Save_As GUI by displaying the filename selected from the list and/or checking if the filename entered already exists.

11. VertStatus.m

This file displays a GUI that prompts the user to check the number of vertices if the value entered is erroneous.

12. bistatic.m

This file displays the Calculate Bistatic RCS GUI.

13. bistat.m

This file implements the functionalities of the Calculate Bistatic RCS GUI by checking all the parameters entered, displaying GUIs with appropriate messages, calculating the bistatic RCS of the selected model, and generating the plot.

14. design.m

This file displays the Design Model Coordinates GUI for encoding the coordinates of the model being designed or edited.

15. dsgn.m

This file implements the functionalities of the Design Model Coordinates GUI by checking the individual coordinates entered and the whole coordinate set.

16. dsgnFN.m

This file clears previous design GUIs and displays a new Design Model Coordinates GUI.

17. dsgnFO.m

This file displays the Open Model File GUI showing the list of existing models which the user may select to display the data set and plot of the geometry.

18. facets.m

This file displays the Design Model Facets GUI for defining the configuration of the component triangular facets of the model.

19. fact.m

This file calculates the factorial of a number n given the parameter.

20. fcets.m

This file implements the functionalities of the Design Model Facets GUI by checking the number of facets, individual nodes and set of nodes, illumination and resistivity values.

21. hlpbistattxt.m

This file contains the text that is displayed in the Calculate Bistatic RCS Help GUI.

22. hlpcoordtxt.m

This file contains the text that is displayed in the Design Model Coordinates Help GUI.

23. hlpfacettxt.m

This file contains the text that is displayed in the Design Model Facets Help GUI.

24. hlpmaintxt.m

This file contains the text that is displayed in the POFACETS Main GUI.

25. hlpmodeltxt.m

This file contains the text that is displayed in the Design Model Geometry Help GUI.

26. hlpmonotxt.m

This file contains the text that is displayed in the Calculate Monostatic RCS Help GUI.

27. hlpbistat.m

This file displays the Calculate Bistatic RCS Help GUI.

28. hlpdsgncoord.m

This file displays the Design Model Coordinates Help GUI.

29. hlpdsgnfacet.m

This file displays the Design Model Facets Help GUI.

30.   hlpmain.m

This file displays the POFACETS Main Help GUI.

31.   hlpmodel.m

This file contains the text that is displayed in the Design Model Facets Help GUI.

32.   hlpmono.m

This file displays the Calculate Monostatic RCS Help GUI.

33.   message.m

This file displays the name of the model selected in the Calculate Monostatic RCS GUI and enables the Calculate RCS button.

34.   mod.m

This file implements the functionalities of the Model Geometry GUI by plotting the model, labeling its vertices and facets, displaying the grid, and printing the data set and model plot.

35.   model.m

This file displays the Model Geometry GUI.

36.   mono.m

This file implements the functionalities of the Calculate Monostatic RCS GUI by checking all the parameters entered, displaying GUIs with appropriate messages, calculating the monostatic RCS of the selected model, and generating the plot.

37.   monostatic.m

This file displays the Calculate Monostatic RCS GUI.

38. mplt.m

This file prints the Monostatic or Bistatic RCS plot.

39. msgComputing.m

This file displays a message that the GUI is calculating either the monostatic or bistatic RCS of the model selected.

40. pofacets.m

This file displays the POFACETS Main GUI where the user may choose to run any of the three modules - Design Model, Calculate Monostatic RCS and Calculate Bistatic RCS.

LIST OF REFERENCES

1.   Jenn, David C., *Radar and Laser Cross Section Engineering*, American Institute of Aeronautics and Astronautics, Inc., Washington, D.C., 1995.

2   Knott, Eugene F., Shaeffer, John F., and Tuley, Michael T., *Radar Cross Section*, 2d ed., Artech House, Inc., 1993.

3.   White, M. O., "Radar cross-section: measurement, prediction and control," *Electronics & Communication Engineering Journal*, vol. 10, Issue 4, August 1998, pp. 169-180.

4.   Knott, Eugene F., Shaeffer, John F., and Tuley, Michael T., *Radar Cross Section Its Prediction, Measurement and Reduction,* Artech House, Inc., 1985.

5.   Moreira, Fernando J. S. and Prata, Aluizio Jr., "A Self Predictor-Corrector Algorithm for Efficient Evaluation of Reflector Antenna Radiation Integrals," *IEEE Transactions on Antennas and Propagation*, vol. 42, No. 2, February 1994, p.246.

6.   Faros, N., *Radar Cross Section Synthesis for Planar Resistive Surfaces*, Naval Postgraduate School Masters Thesis, December 1994.

7.   Waddell, J. M., *Scattering from Rough Surfaces*, Naval Postgraduate School Masters Thesis, September 1995.

8.   Jenn, David C., Physical Optics Applied to Triangular Facets, Unpublished Notes, Naval Postgraduate School, Monterey, California, March 2000.

9.   "AIM-9 Sidewinder."
      [http://www.af.news/factsheets/AIM_9_Sidewinder.html]. November 1999.

10.   Ruck, George T., and others, *Radar Cross Section Handbook*, v. 1, Plenum Press, New York, 1970.

11.   Enander, Eva Pärt, and Sjöberg, Anders, *The MATLAB®5 Handbook*, Addison-Wesley, Harlow, England, 1999.

12.   Marchand, Patrick, *Graphics and GUIs with MATLAB*, 2d ed., CRC Press, New York, 1999.

13.   Mathworks, *MATLAB Handling Graphics*.

14.   Mathworks, *Building GUIs with MATLAB*, version 5, 1984-1997.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

No. of Copies

1. Defense Technical Information Center 2
   8725 John J. Kingman Rd., STE 0944
   Ft. Belvoir, Virginia, 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, California, 93943-5101

3. Chairman, Code EC 1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California, 93943-5101

4. Professor David Jenn, Code EC/Jn 3
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California, 93943-5101

5. Professor D. Curtis Schleher EC/Sc 1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California, 93943-5101

6. Ronald Radlinski 1
   Office of Naval Research
   800 N. Quincy St., Ballston Tower One
   Arlington, Virginia, 22217-5660

7. Dante E. Garrido 2
   Department of Physics
   AMA Computer College
   Pasig, Metro Manila
   Philippines

8. CDR Elmo E. Garrido, Jr. 2
   AC of NS for Education and Training, N8
   Philippine Navy
   Roxas Boulevard, Manila
   Philippines